ARKITE

Software Manual

Version 2023.3

About this document

Purpose of the document

This document aims to enable both new and experienced users of the Arkite software to translate their standard operating procedures into a project and as such configure their workstation. The manual contains an overview of all basic and advanced software features.

Note

- *For instructions on installing and mounting the sensor, please refer to the **Arkite hardware installation guide**.*
- *For instructions on installing the Arkite software, please refer to the **Arkite software installation guide**.*
- *This document describes the features that are available in the 2023.3 version of the Arkite software.*

Main changes

The table below lists the main changes in the software (and therefore also in this document) compared to the previous version 2023.1.

Area of the Change	**Description**	**Relevant section in the document**
Connectivity	– A new **DMX** communication module has been added. With this module, you can connect DMX lights through an Enttec Open DMX USB interface (SKU 70303) – **Database communication:** The result of a database query containing multiple rows can now be stored as a JSON array. – **OPC UA communication**: It is now possible to trust server certificates from within the Arkite software.	Section **DMX** on page 167 and **6.3.4 Configuring lights for a DMX communication module** on page 179
User Management	It is now possible to set up authentication with Active Directory in the Arkite software. Access and permissions can be assigned to specific Active Directory User Groups.	Chapter **16 User management** on page 241
Conditions	The **Conditions** feature has been expanded: It is now possible to apply inverted logic when setting a condition. You can not only indicate to execute a step when a condition is true, but also when it is false.	Section **5.7 Working with conditions** on page 144
Process Steps	– Repetition of composite steps: you can now indicate a composite step should be repeated a specific number of times, or for each element in a list or JSON array. – A new **Switch Language** process step has been added that allows you to change the language of your text instructions without changing the user. – The **Set Variable** step has been enhanced so you can set a variable based on its name. The variable name can be fixed or dynamic. – The **Process Control** step has been enhanced so that the process that is controlled can now also be dynamic, i.e. based on the value of a variable. – You can now add generic detection steps in which the detection that is activated depends on the value of a variable.	Section **5.4 Configuring process steps** on 104
Variables	In all properties with dynamic content, it is now possible to use a variable within a variable.	Throughout the document

Table of Contents

1. About the Arkite software

1.1 Purpose of the Arkite software

Arkite offers the ultimate operator guidance platform. It transforms workstations into a digital and interactive environment.

The Arkite software allows you to configure the platform and adapt it to your specific working environment. Through the Arkite software, you can translate your standard operating procedures into a project without any programming knowledge.

Three license types are available for using Arkite's Operator Guidance platform:

- **Arkite Guide** license:

  Once your project has been configured, Arkite's operator guidance platform will be able to provide your operators with real-time picking and assembly instructions through Augmented Reality.

  The system cannot verify if a process step was done correctly. The operator will always have to confirm he has executed step and press the **Next** button on the touch screen, a hardware button or a foot pedal to continue to the next step.

  All features from this license are also included in the **Arkite Validate** and **Arkite Vision** licenses.

- **Arkite Validate** license:

  When using this license, your Arkite Operator Guidance Platform is equipped with an Arkite 3D sensor. The system will verify if a process step is executed correctly. It will automatically continue to the next step after a step has been done correctly, and will warn your operators in case of any error.

  All features from this license are also included in the **Arkite Vision** license.

  Throughout this document, features that are only available when a workstation is used in combination with a sensor are tagged with a note **'Validate and Vision license only'**.

- **Arkite Vision** license**:**

  When using this license, your Arkite Operator Guidance Platform is equipped with an additional high definition infrared vision sensor. You can then benefit from Arkite's smart vision features.

  - Product inspection during assembly
  - Improved presence detection for small objects (> 0,5 cm)
  - Object orientation
  - Tool tracking

  Features that are only available when a workstation is used in combination with an Arkite Vision sensor are tagged with a note **'Vision license only'**

  Tip

  For an overview of different applications of Arkite's operator guidance platform, have a look at the references on our website:

  https://arkite.com/references/.

1.2 How does the Arkite 3D sensor work? (Validate and Vision license only)

Technology

The Arkite 3D sensor used by Arkite's operator guidance platform is a Time-of-Flight sensor. Combining this ToF-sensor with intelligent algorithms, the Arkite software can verify the presence of objects and validate operating procedures.

The sensor uses two types of streams to capture data:

- Depth stream
- Infrared stream

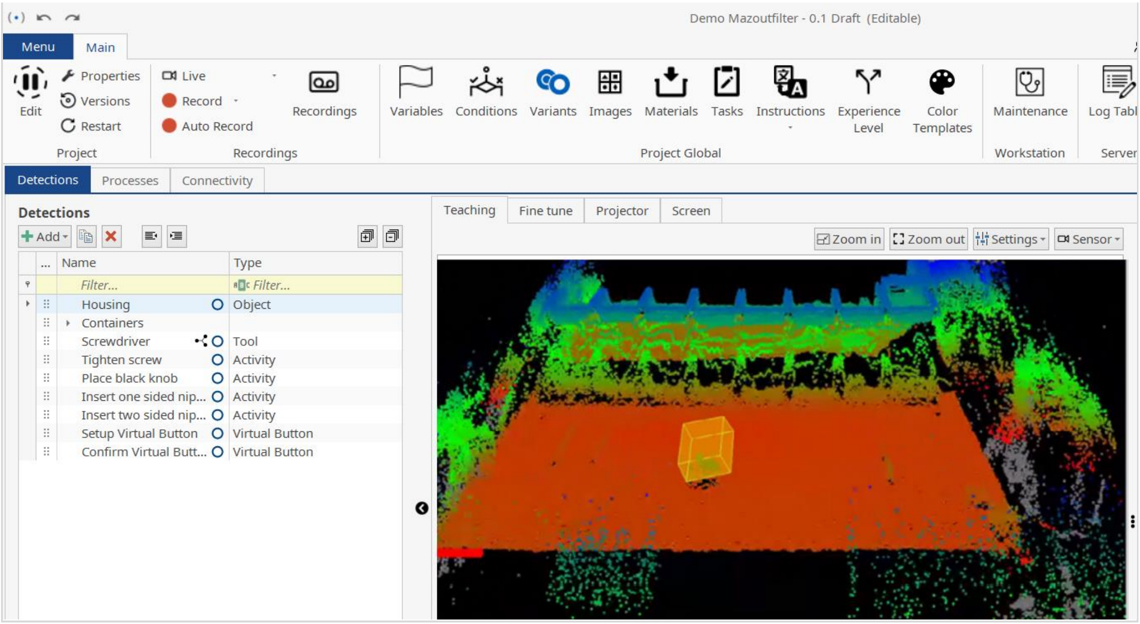
By default, the Arkite software will automatically select the stream with the best contrast. However, you can overrule this selection and force the software to use a specific stream.

Note

If the depth and IR stream do not provide sufficient contrast, it's also possible to use the RGB image. If you would like to try using this stream, please contact Arkite for support.

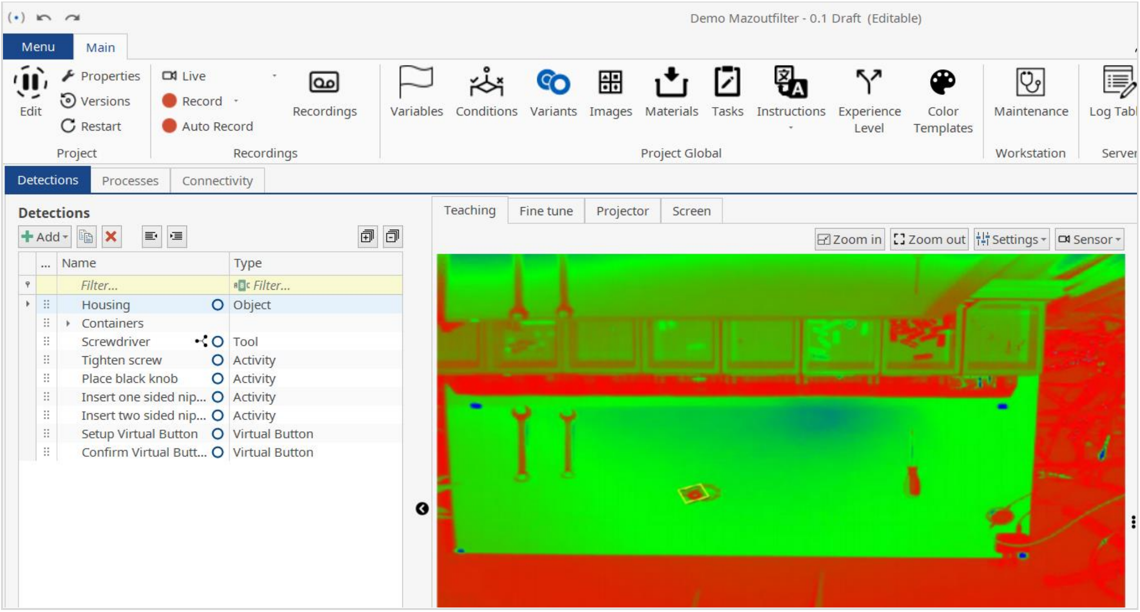
Depth stream

The sensor is sending out infrared light, invisible to the human eye. The IR light reflects on all objects. For each pixel, the time is measured for the light to travel from the sensor to the object and back. Then the distance of each pixel to the sensor is calculated. This results in a 3D image or point cloud. Virtual boxes are created on top of this real-world 3D image. The activity of an operator or the presence of an object in the box will lead to a detection trigger. By combining all these triggers in the correct sequence, the Arkite software can validate operating procedures.



Infrared stream

The reflected infrared light beams will also create an infrared image. The reflection intensity of each pixel is measured. The more IR light is captured by a pixel, the higher the value will be. A matte black surface will return a near 0 value, a highly reflective surface will return values up to 9000. This results in a 2D image containing distinctions of color and material of objects.



A specific application of the infrared stream is edge detection. When using this application, the algorithm will specifically look for edges in the infrared image to validate the presence of a certain object. This detection is mainly used for tooling.

1.3 How does the Arkite Vision sensor work? (Vision license only)

What?

When you are using Arkite's Operator Guidance Platform with a Vision licence, the platform is equipped with an additional high-definition infrared Vision sensor.

How does it work?

The Arkite Vision sensor captures high-quality image and transmits the image data to the Arkite PC over a GigE data interface. The Arkite software processes the images and applies different algorithms to them to achieve accurate and efficient image analysis that can be used for the smart vision features.

1.4 Installing the Arkite software

What?

The Arkite software is split in a workstation and a server component. The server component, containing all data and resources for multiple clients and projects, should preferably run on a separate windows server to obtain higher security, maintainability and data integrity.

Installation instructions

For instructions on installing the Arkite software, please refer to the **Arkite software installation guide**.

1.5 Starting the Arkite software for the first time

What?

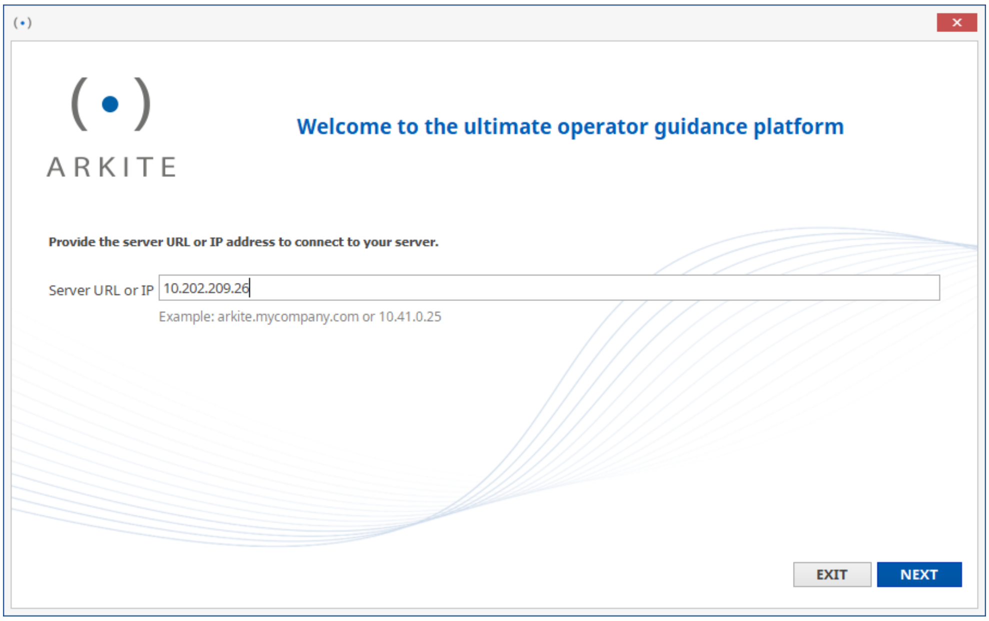
The Arkite software automatically starts after the workstation has been turned on. If it does not, you can double-click the Arkite icon on the desktop.

Connect to the server

When you open the Arkite software for the first time, a welcome wizard appears.

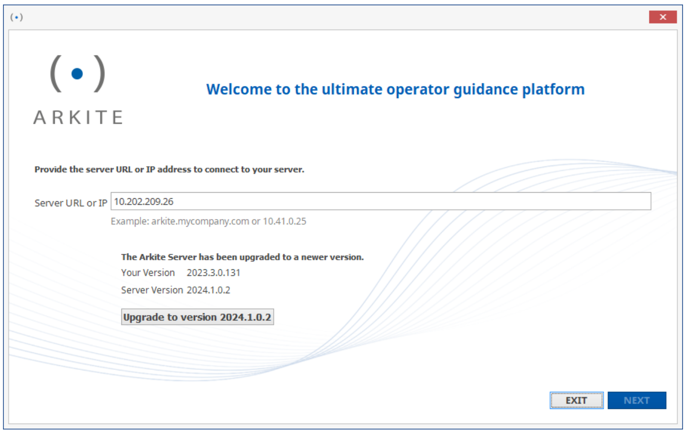
In first screen of this wizard:

1. In the **Server URL or IP** field, enter the URL of IP address of the server where the server component was installed.
2. Click **Next**.



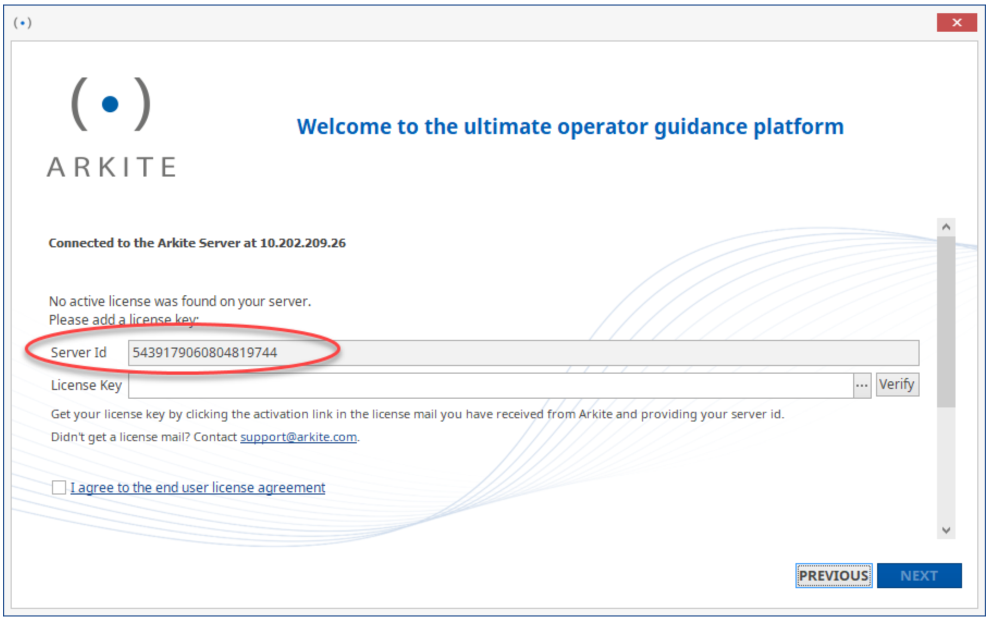
Note

- *If the server component is installed on your workstation, fill in **localhost**.*
- *If the software version on your workstation is not the same as the one on your server, you will first need to upgrade. A button will be shown allowing you to download and install the latest software version.*



Enter your server license key

If your workstation is the first one to connect to the server, you will need to provide a license key in the next screen of the wizard.



Note

If another workstation already connected to the server before and provided the server license key, this step won't appear. You will immediately be taken to the next step in the wizard.

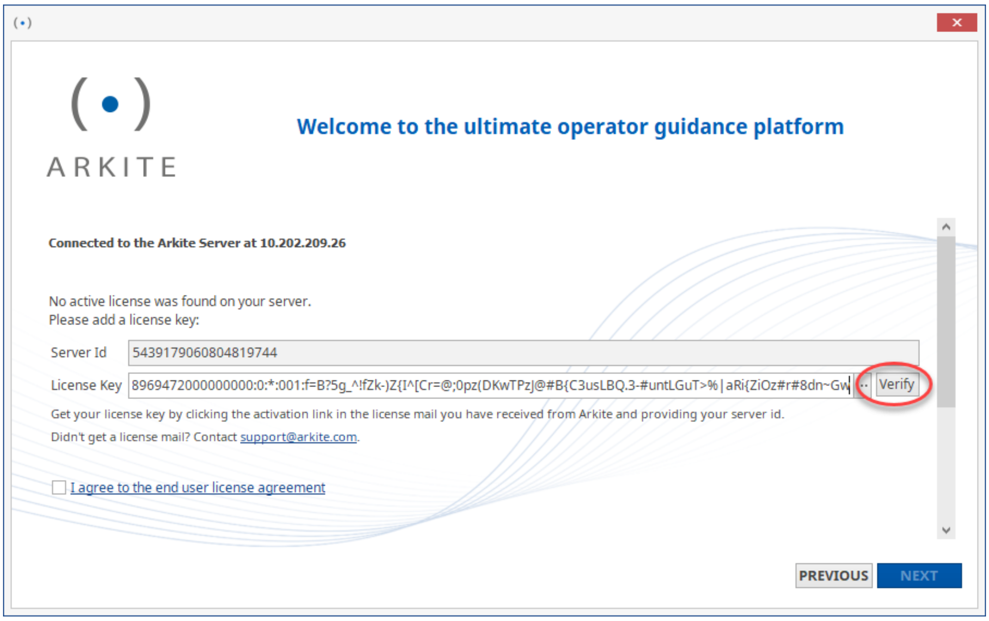
To obtain your server license key, click the activation link you have received in an e-mail from Arkite, fill in the webform and click **Submit**. If you haven't received an activation link, please contact support@arkite.com.

You will need to provide your **Server Id** (you can copy this from the information in the screen in the wizard).

After a few minutes, a license key will be generated and sent to your e-mail. Once you have received this e-mail:

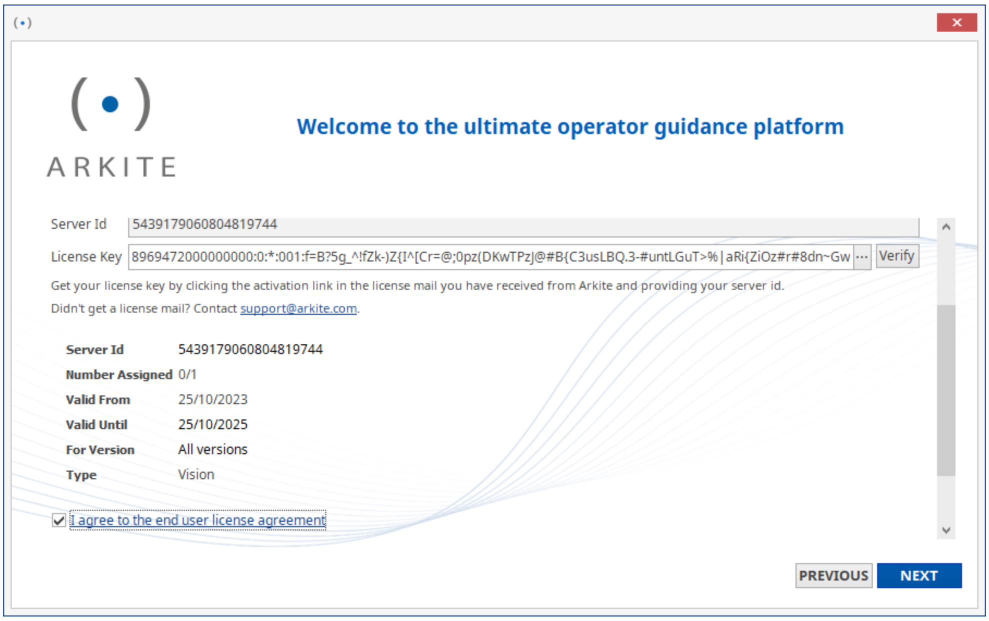
1. Copy the license key from the e-mail or download the **.key** file in attachment of that e-mail and save it on the workstation.
2. In the wizard, paste the license key in the **License Key** field or click the **...** button next to the field and select the **.key** file you have just saved.

3. Click **Verify**.



The information related to your server license appears. You will see:

- The **number of workstations** the license can be used for, and how many of those have already been assigned to the license
- The **validity** of your license
- The **software version** for which the license is applicable
- The **type** of your license: **Guide**, **Validate** or **Vision**.



4. Check the **I agree to the end user license agreement** checkbox.
5. Click **Next**.

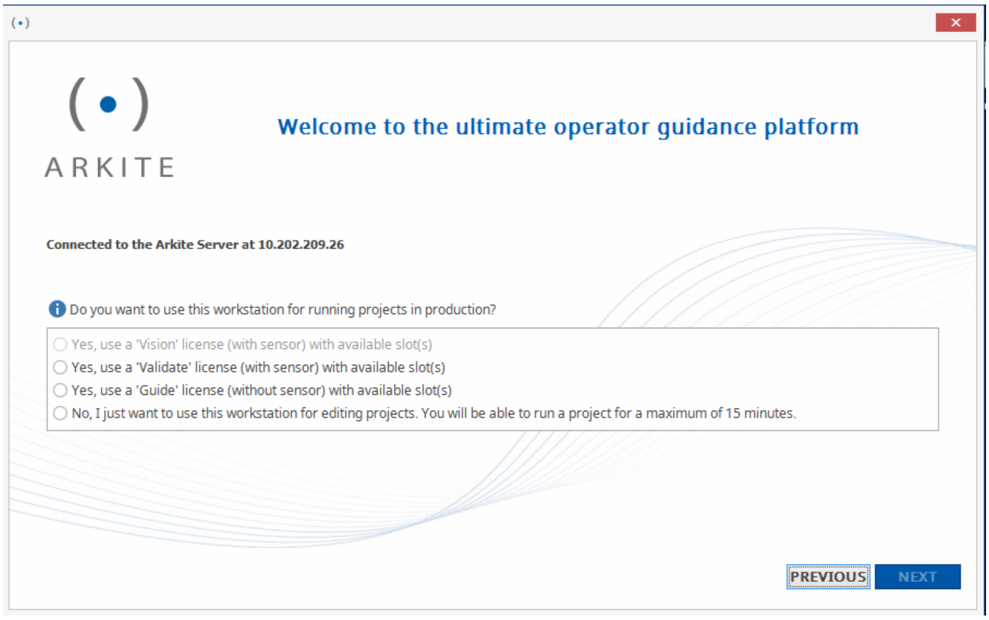
Note

- *In case you have a perpetual license, your license key is version specific. When you install a new version of the Arkite software, you will have to request a new license key. Recurring licenses can be used for any version of the software.*
- *Between three and two months before your license expires, you will receive a quotation from Arkite for the renewal of your contract. Once you have placed the purchase order, you will receive a new license key. You can then add this new license key as described above, without assigning any workstations to it yet. For more information, consult the section* ***How to renew a license key*** *on page 211.*
- *Your Arkite software license is linked to the server on which you install the server component. In order to move your Arkite installation to a different server while retaining your license, you should execute the steps as described in the procedure* ***How to move your Arkite installation to a different server****.*

Assign your workstation to a license

In the next step of the wizard, you can assign your workstation to a license, on the condition that a license key with available slots is present for the server you are connected to. If license keys for different license types are available, you will be able to choose which license type to use.

Indicate if you want to assign your workstation to a license and click **Next**.



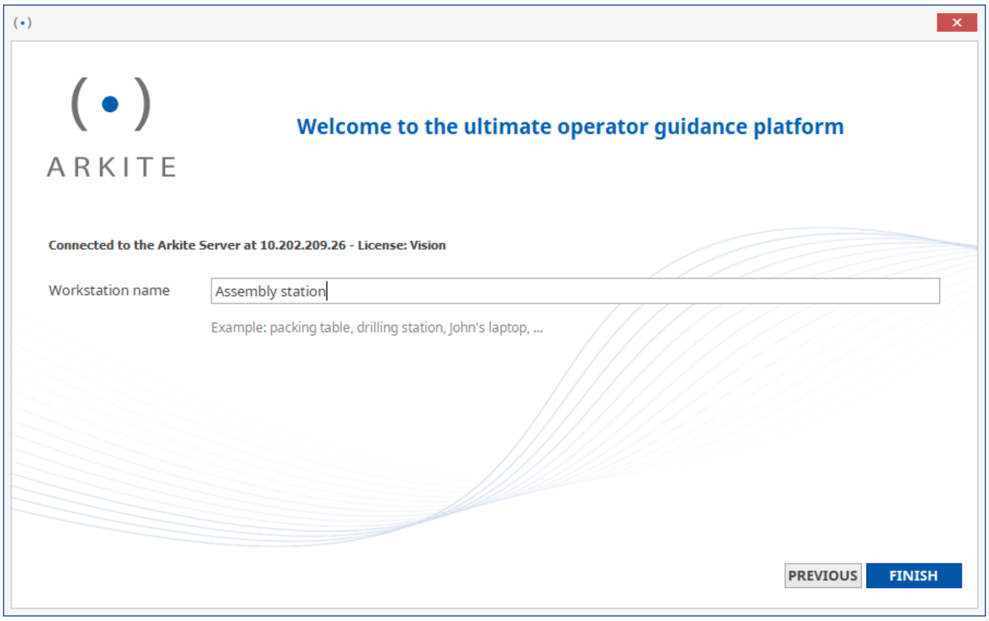
Note

- *If you have a Vision license, you will only be able to add workstations to it that are* ***Vision Ready****. If your workstation is not* ***Vision Ready****, the option will be disabled. In that case, finish the installation without assigning your workstation to a license. Then follow the steps described in the procedure* ***How to make your workstation Vision Ready*** *(available on the customer drive). The number of workstations you can make* ***Vision Ready*** *is limited to the number of workstations for which you have bought a Vision license. If you already have reached this number, your request will be denied.*
- *If your workstation is* ***Vision Ready****, you will not be able to assign it to a* ***Validate*** *or* ***Guide*** *license.*

- *If no licenses with open slot(s) are available or if you choose to not assign your workstation to a license, you can use your workstation as an **editor**. You will be able to prepare and configure projects on them, but you cannot use them to run a project. For testing purposes, you will be able to run your project for a maximum of 15 minutes. This way, you can prepare your project on your laptop at your desk.*
- *When your workstation is an editor, you can also edit a project for an another workstation connected to the same server. To do so, you can use the **Switch Workstation** feature. This way, you can limit the production downtime on your workstation as a minimum. For more information, refer to section **3.6 Switching workstations** on page 55.*
- *At any moment, you can add or remove license keys and change the workstations that have been assigned to them. For more information, refer to section **10.4 Managing licenses** on page 208.*

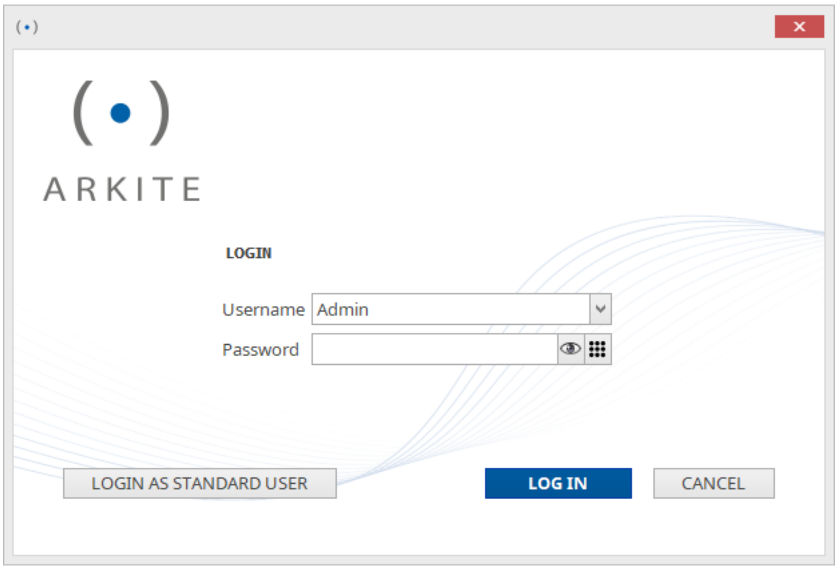
Provide a name for your workstation

In the next step of the wizard, enter a name for your workstation in the **Workstation name** field and click **Next**. Choose a name that will make it easy for you to identify your workstation.



Log in

A login dialog appears.

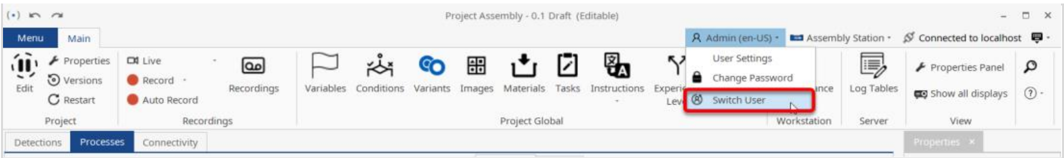


Provide the password for the **Admin** user. By default, this is 'Arkite3600'. Next, click **Login**.

Note

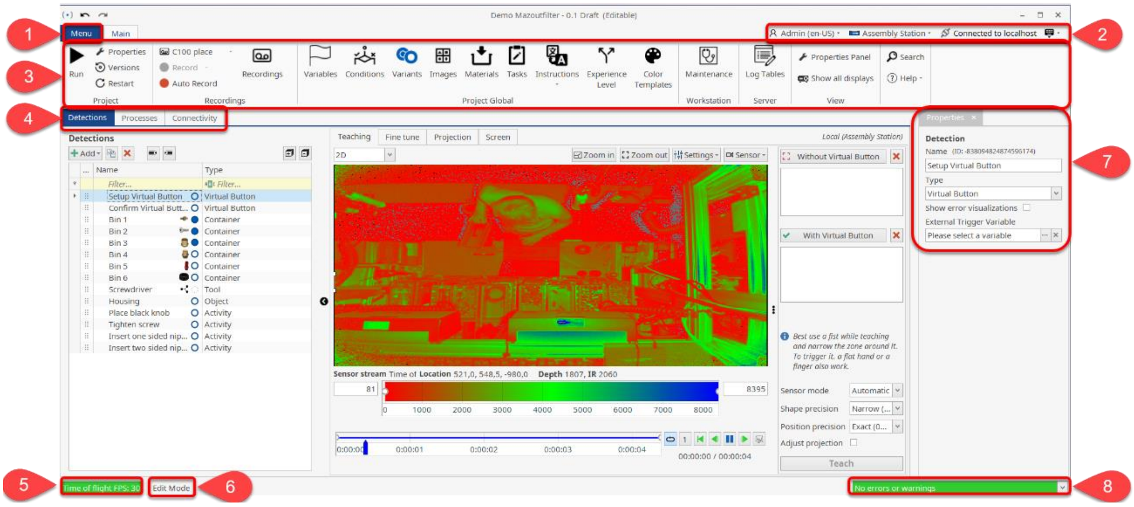
- *You can click the button to make the password you've entered visible and verify if you haven't made any typing mistakes.*
- *You can click the button to display an online keyboard, in case you are working with a touch screen for example.*
- *If you do not provide a password for the Admin user within 8 seconds, you will log in as standard user with read-only access to the Arkite software.*
- *Once you have logged in with the Admin user, you can set up other users and roles at your convenience. For more information, see chapter **16 User management** on page 241.*

To login with another user, click your username in the top bar and select **Switch User**.



1.6 Arkite User Interface Overview

The image below gives an overview of the Arkite software user interface:



1. The **Menu** tab gives access to features that apply to the Arkite software as a whole:
   - The overview of projects
   - Server Settings
   - Dashboarding
   - Workstation Overview
   - User management features
   - Workstation calibration
2. The **top bar** provides you with information about:
   - The user that is currently logged in
   - Your workstation's name
   - The server you are connected to

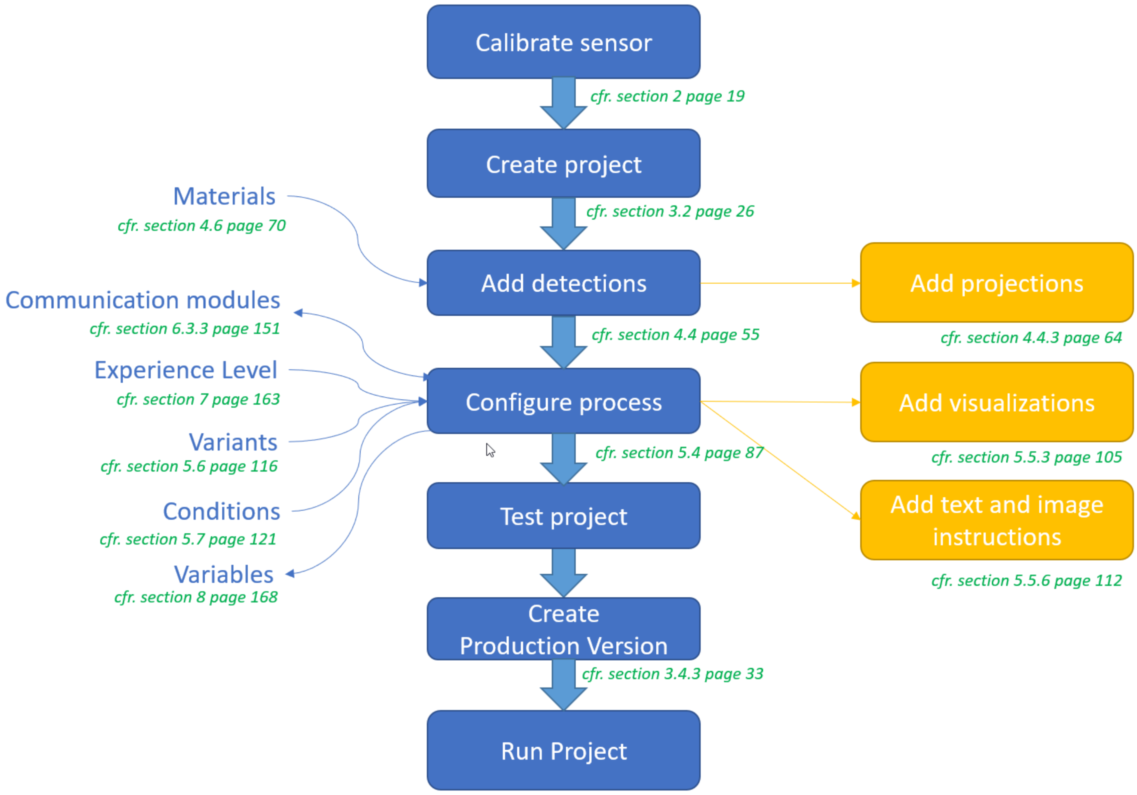
   It also allows you to switch user or workstation (in case you are logged in as an editor) or update your server connection settings.
3. The **Project** ribbon gives access to features that apply to the project as a whole or are used across the project, e.g.:
   - Project properties
   - Versions
   - Recordings
   - Variables
   - Variants and conditions
   - Experience Levels
   - Sensor calibration

- Help files

4. The **Main** navigation gives access to the features that are key for building your project:
   - Detections
   - Processes
   - Communication modules
   - Displays
5. The **Sensor status** indicates the status of your main sensor.
6. The **Project mode** indicates whether your project is in **Run** or **Edit** mode.
7. The **Properties** pane shows the properties of the component that is currently selected.
8. The **Project warnings** indicate whether there are any errors or warnings into account.

1.7 A typical process flow

The image below gives you an idea of how a project is typically set up. Refer to the sections mentioned in the image for more information.

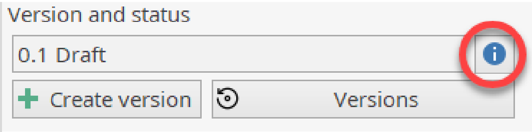


1.8 Help and Support

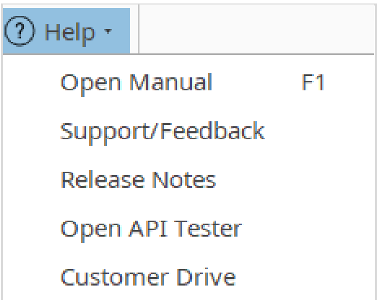
Help in the Arkite software

You can find a lot of help within our software:

- For many features, you will see a button. Clicking this button will give you specific information for that feature.



- In the ribbon, you will see a **Help** button. This button gives access to this manual, the release notes of the software and the Open API Tester page. It also contains a link to the Customer Drive where you can find more extended documentation.



Note

*For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*

Contact our support team

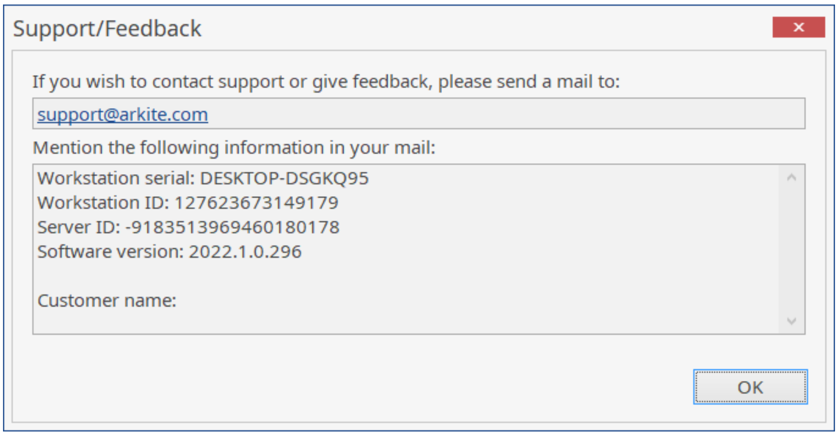
If you still have questions after reading this manual, our team is happy to support you.

To obtain support, you can:

- Create a ticket on our portal
- Send an e-mail to support@arkite.com in which you mention:
  - your Workstation Serial Number
  - your Workstation ID
  - your Server ID
  - the software version you are using
  - your customer name

Tip

If you have trouble finding this information, click the ? ***Help*** *button in the ribbon and select* ***Support/Feedback****. A window with the required information will appear.*



Our team will get back to you as soon as possible.

TeamViewer

On every workstation delivered by Arkite, TeamViewer has been installed. This software allows our support people to access the workstation and provide you with direct support when needed.

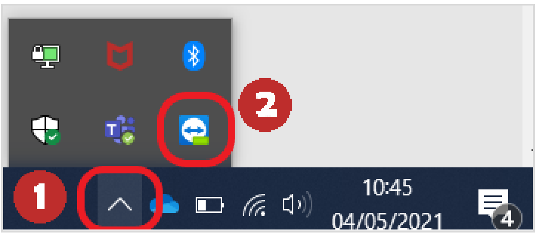
TeamViewer automatically starts up when the workstation is turned on.

If you are using the Arkite software your own workstation, you also need to install Team Viewer using the link below:

https://www.teamviewer.com/download/windows/

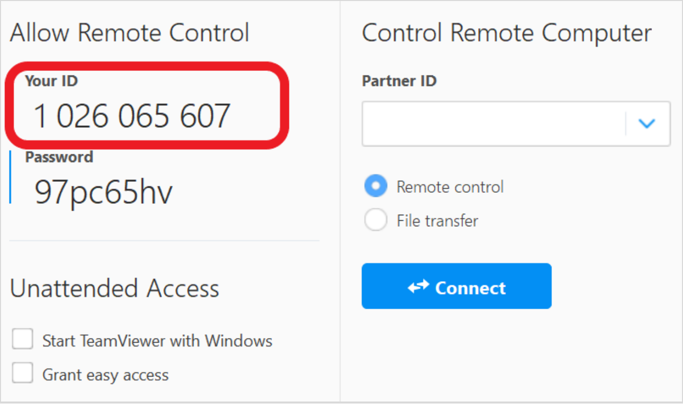
To open TeamViewer:

1. Click the upward arrow in the lower right corner of your screen.
2. Click the TeamViewer icon.



TeamViewer opens.

3. Provide our support team with your ID. Make sure the workstation is connected to the internet.



Note

You can also use TeamViewer yourself in order to take control of the workstation remotely. You can install TeamViewer using the link below:

https://www.teamviewer.com/download/windows/

After the installation, enter the TeamViewer ID and password of the workstation.

2. Calibration

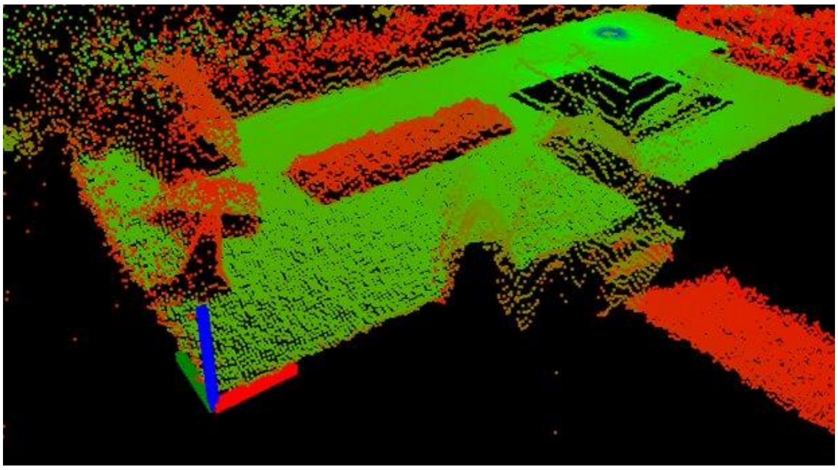
2.1 About calibration

Before you can start configuring your project, you will have to calibrate your workstation. Calibration defines the coordinate system (mm) and origin for all detections.

If you are using the system with Arkite 3D sensor (**Validate** or **Vision** license), both the sensor and the projector need to be calibrated.

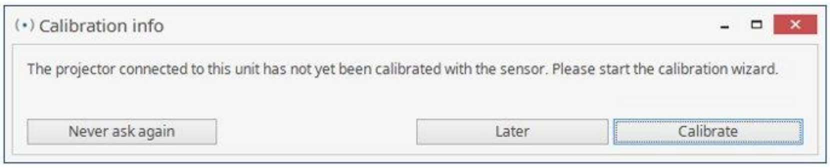
When using a Vision license, you will additionally need to calibrate the **Arkite Vision** sensor.

During sensor calibration, each XYZ coordinate is translated to a pixel in the sensor image with a corresponding RGB value.



Projector calibration defines the coordinate system that is used to define where projections need to be shown on the workbench.

If calibration has not taken place yet on a particular workstation, a popup will appear.



Calibration is one-time operation, but after the calibration the projector and, if available, the sensor need to remain fixed relative to the workbench. When the sensor and/or projector have moved, you have to recalibrate.

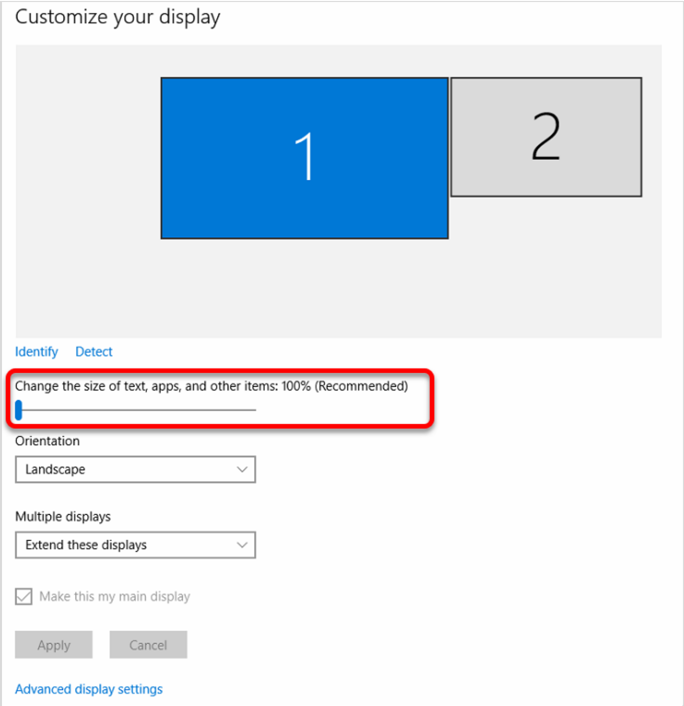
2.2 Calibrating your workstation

2.2.1 Calibrating a workstation with Arkite 3D sensor using the calibration wizard (Validate and Vision license only)

What?

The Arkite software contains a calibration wizard that will take you through all the required steps to calibrate your sensor and projector.

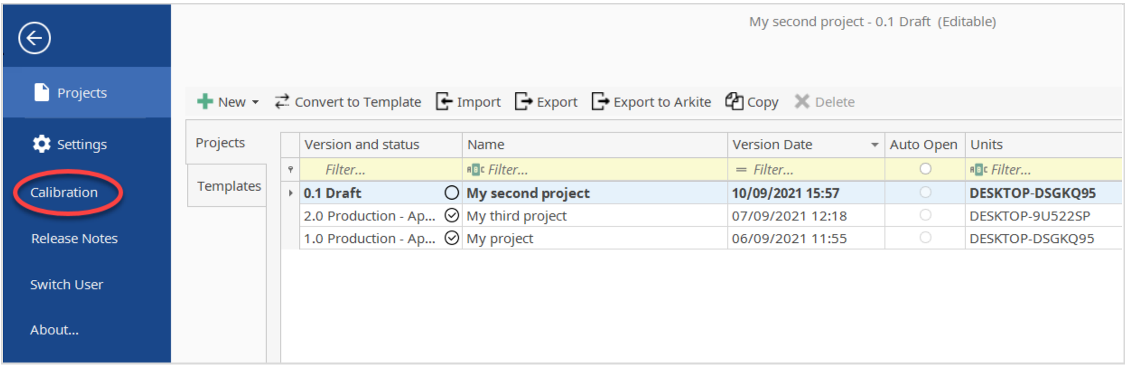
> ***Note***
>
> *For an optimal result, the scaling settings for your projector in Windows should be configured at 100%. Please verify before you calibrate. In order to do so, right-click in your desktop and select* ***Display Settings****.*



Steps

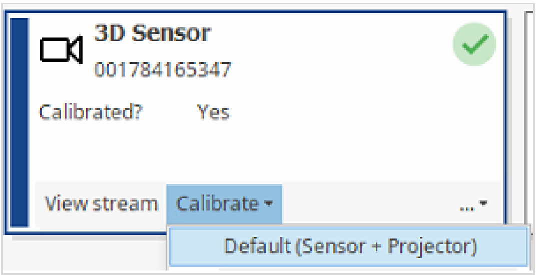
To use the calibration wizard, execute the following steps:

1. If calibration has not taken place yet on your workstation, click **Calibrate** in the **Calibration info** popup.
   When recalibrating, click the **Menu** tab and then **Calibration**.



Note

*When you are in a project, you can also navigate to the **Connectivity** tab, select the **3D sensor** and select **Default Calibration (Sensor+Projector)** from the **Calibrate** menu.*



*When you have created the project before calibrating, a **Projector** display is added with the screen resolution of the device you are working on (e.g. your laptop). This may not match with the actual projector resolution, used during calibration. This will cause projections to be faulty. You can fix this by updating the navigating to the **Connectivity** tab, selecting the **Projector** card and updating the **Resolution** property to the correct value in the **Properties** pane.*

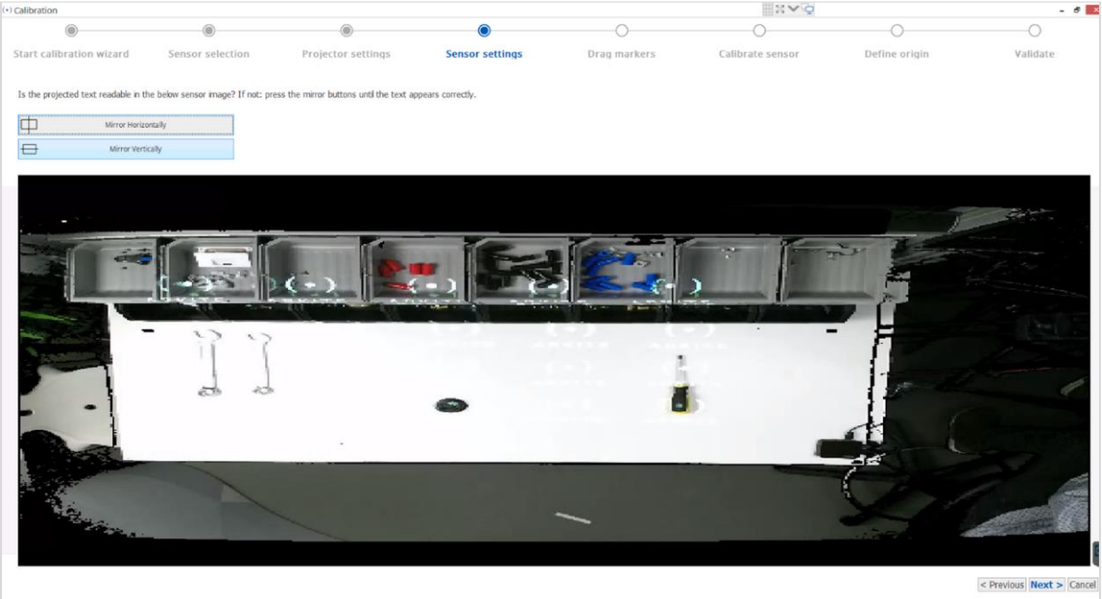
2. For the initial calibration, select **Fully Restart Calibration** and click **Next**.
   To calibrate additional sensors or projectors, use the other options.
3. Select the **3D Sensor** and click **Next**.
4. Check the projector settings:
   - Verify if the Arkite logo is projected. If not, change the output to which the projector is connected.
   - Verify if the Arkite logo is projected in the correct orientation on the workbench. Rotate and/or mirror if required.
   - Optionally, check the **Enable Height** checkbox.
     That is useful when you have many detections above the calibration surface, or if your detections are dynamically updated by an external system.

Note

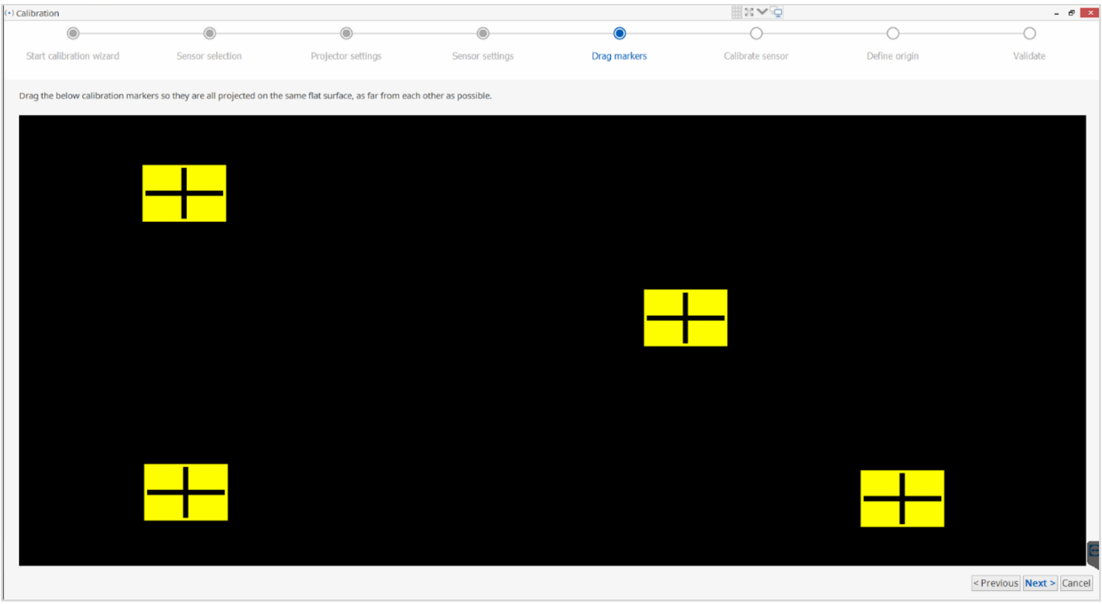
*Currently, the **Enable Height** option is only available for Optoma ZH420, Optoma ZW400 and Optoma ZH403 projectors.*

- When finished, click **Next**.

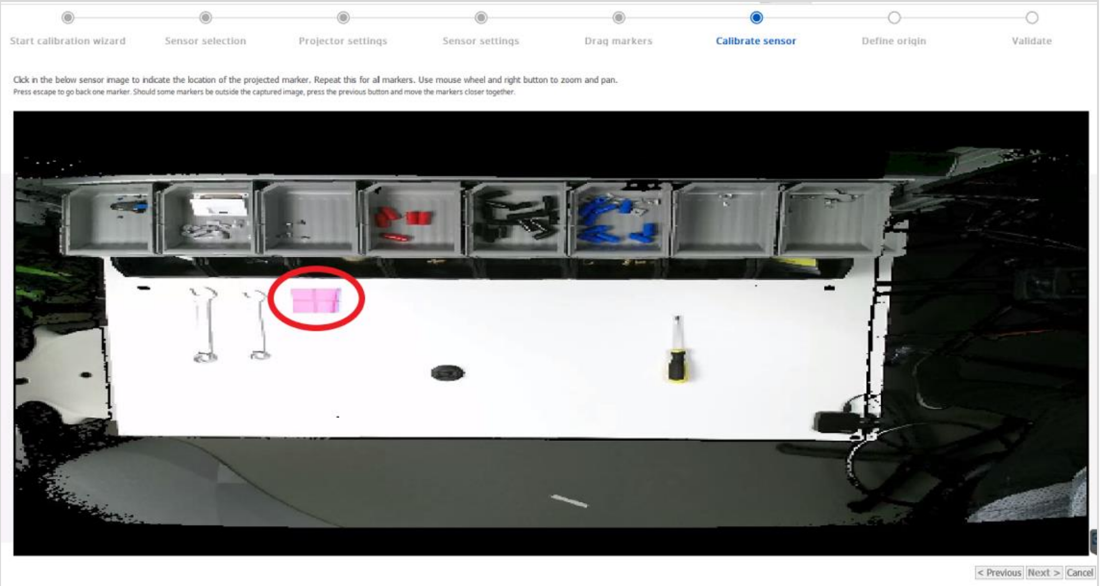
5. Make sure the sensor image matches the orientation on the workbench. Apply mirroring if required. When finished, click **Next**.



6. Verify whether the four calibration markers are projected correctly on the workbench. Drag them into a position where they are on a flat surface (i.e. the table, not a tool or container), as far from each other as possible.

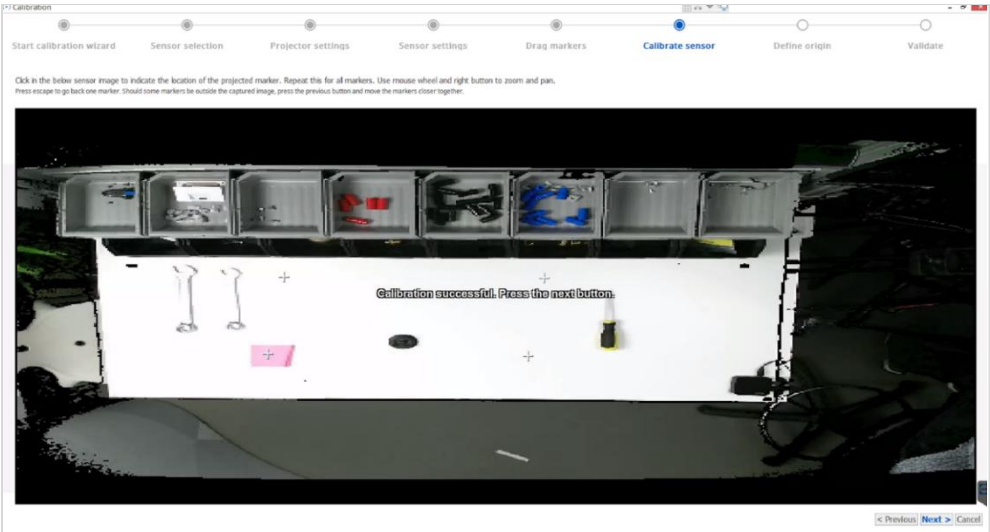


7. Click the center of each marker in the projected image.
   Be as precise as possible. Use the mouse wheel and the right mouse button to zoom and pan.



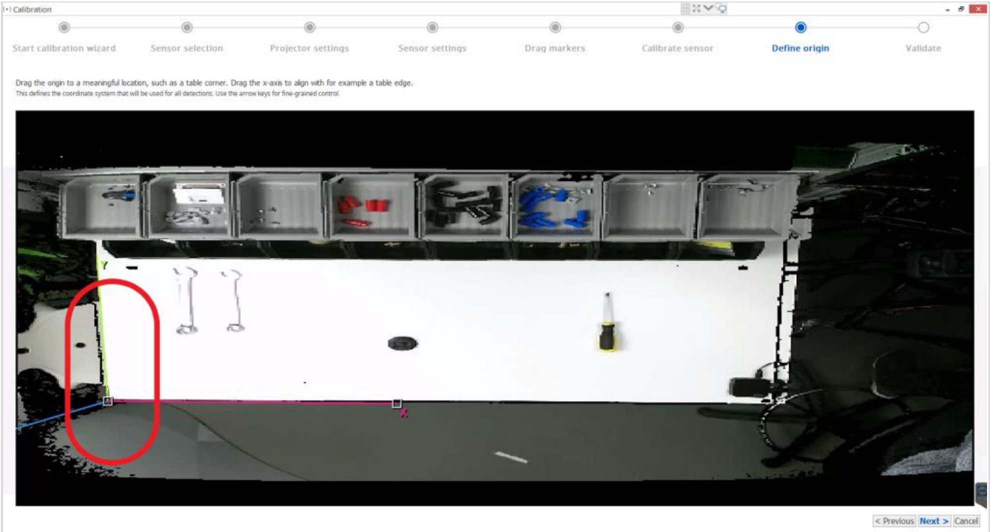
Tip
If a marker is difficult to see, you may add a piece of paper on that location.

8. If calibration was successful, this will be indicated on the screen. Click **Next**.



Note
In case calibration was not successful, a message will appear indicating the calibration is invalid. If the calibration is invalid because of one or more calibration marker(s) for which no depth value could be measured, the respective marker(s) will be indicated by a red circle. Click the ***Escape*** *button once or more until you reach the invalid marker. Click the marker(s) again. If that doesn't work, click* ***Previous*** *to return to the previous step and drag the markers to another location.*

9. Define the origin of your image by dragging it to a meaningful location, e.g. the corner of the workbench. Click **Next**.



Note

With the workstation, you will also receive a small aluminum plate with the Arkite logo. You should screw the plate onto the workbench at the location where you want define your origin. This will help you to calibrate, or to re-calibrate in case of issues.

10. Verify if the projection on your workbench matches the position on the sensor image. If it does, click **Finish**. If not, click **Previous** to adjust your calibration.

Result

Your sensor and projector have been calibrated. If you have multiple sensors or projectors, run the calibration wizard again to calibrate them as well.

2.2.2 Calibrating a workstation with Arkite Vision sensor (Vision license only)

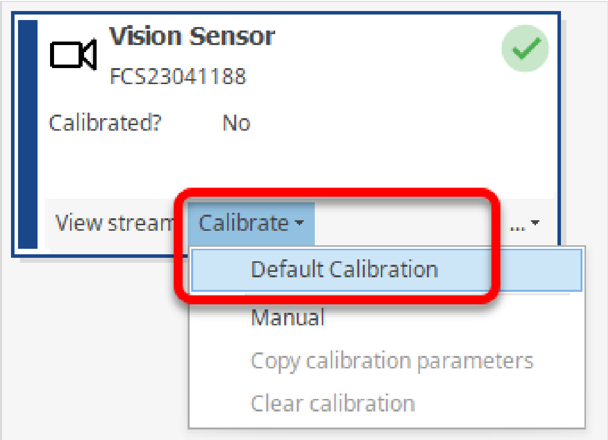
What?

When you are using the Arkite Operator Guidance Platform with a Vision license, also your Vision camera will need to be calibrated.

Steps

Follow the steps in the Arkite Vision sensor calibration wizard. When using your workstation for the first time, the calibration wizard will pop up automatically after you have finished the Arkite 3D sensor and projector calibration wizard.

If not, you can start the wizard by navigating to the **Connectivity** tab, select the **Arkite Vision Sensor** and select **Default Calibration** from the **Calibrate** menu.

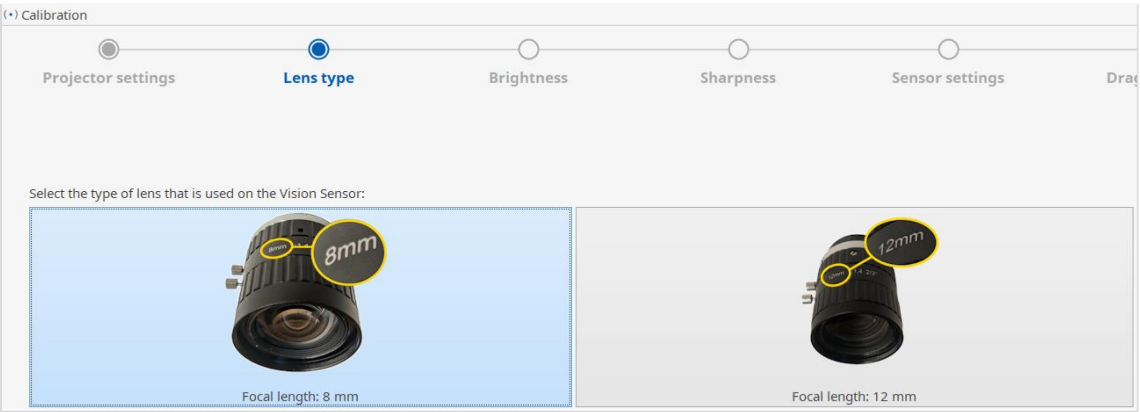


To use the calibration wizard, execute the following steps:

1. Check the projector settings:
   - Verify if the Arkite logo is projected. If not, change the output to which the projector is connected.
   - Verify if the Arkite logo is projected in the correct orientation on the workbench. Rotate and/or mirror if required.
   - Click **Next**.
2. Select the type of lens that is used on your Arkite Vision Sensor and click **Next**.

   Note

   You can find this information on the lens.

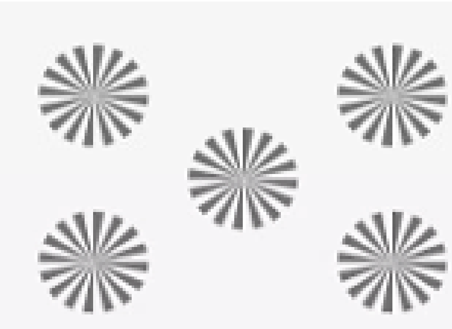


3. Adjust the brightness of the Arkite Vision sensor:
   - Put a typical workpiece under the sensor.
   - Indicate the area of interest dragging the white rectangle.

- Rotate the diaphragm ring (=the top ring) of the Vision sensor until the status is green.

- Tighten the screw.
- Click **Next**.

4. Print 5 papers with Siemens stars and lay them out in the order below on the workbench.
Rotate the focus ring (=the bottom ring) of the Vision sensor until the image is sharp.



Note
You can download a document with a Siemens star from our software or from the customer drive.

5. Verify whether the four calibration markers are projected correctly on the workbench. Drag them into a position where they are on a flat surface (i.e. the table, not a tool or container), as far from each other as possible.
6. Verify if text is projected in the correct, readable orientation on the workbench. Rotate and/or mirror if required.

Note
You can download a document with large text from our software or from the customer drive.

7. Click the center of each marker in the projected image.
8. Because the Arkite Vision sensor uses infrared, the markers are by default not visible in the captured image. Therefore, print a paper with crosshair markers and align them with the projected markers to make the visible in the captured image.
Be as precise as possible. Use the mouse wheel and the right mouse button to zoom and pan.

Note
You can download a document with crosshair markers from our software or from the customer drive.

Result

Your sensor and projector has been calibrated.

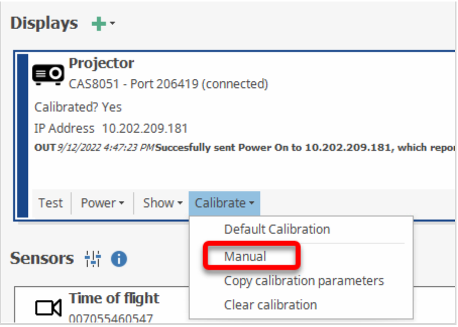
2.2.3 Calibrating a workstation without sensor (Guide license)

What

If you are using the system without sensor (**Guide** license), only the projector needs to be calibrated by entering coordinates in mm.

Steps

1. Open a project.
2. Navigate to the **Connectivity** tab.
3. Select the **Projector** card.
4. In the **Calibrate** menu, select **Manual**.



5. Follow the wizard and enter the coordinates of the markers in mm.

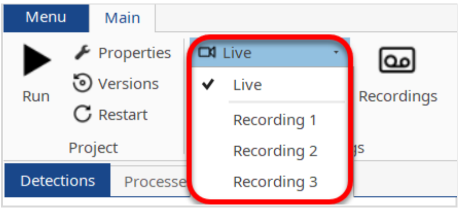
2.3 Other calibration options

Apart from the calibration wizard, a few other options are available in the **Calibration** menu of the **3D Sensor, Vision Sensor** sensor and **Projector** cards on the **Connectivity** tab:

- **3D Sensor:**
  - **Without projector:**
    Use this option to calibrate the sensor only.

    Tip

    When adding detections at your desk based on a recording, first select a recording and use this calibration option.



  - **Manual:**
    Calibrate a sensor by entering coordinates in mm.

- **Vision Sensor (Vision license only):**
  - **Default Calibration:**
    Calibrate the Vision Sensor based on an already calibrated projector that covers the same area.
  - **Manual:**
    Calibrate a vision sensor by entering coordinates in mm.
- **Projector:**
  - **Default Calibration:**
    Calibrate the projector based on an already calibrated 3D sensor that covers the same area.
  - **Manual:**
    Calibrate a projector by entering coordinates in mm.

Note

- *To delete the calibration of a sensor or projector, click the **Calibration** menu on the corresponding card and select **Clear calibration**.*
- *To consult the parameters that have been used during calibration of a sensor or projector, click the **Calibration** menu on the corresponding card and select **Copy calibration parameters**. You can then paste them in a text editor.*

3. Projects

3.1 About projects

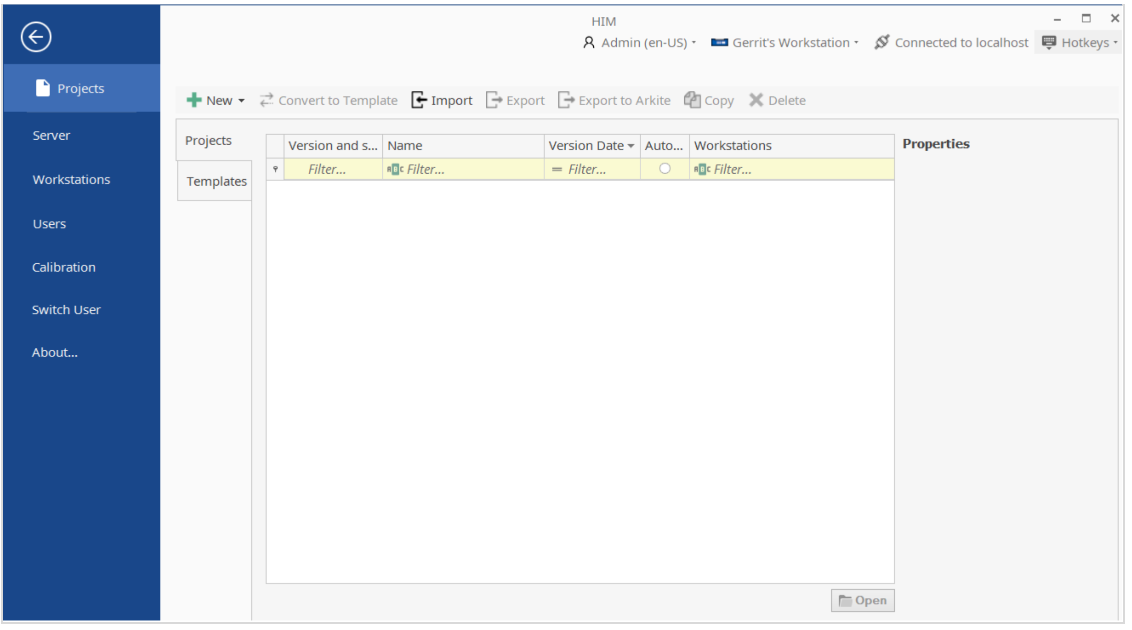
Projects contain all the information that is required to support your assembly process. Typically, a project is created for every product, while product variants are managed within the same project. It is perfectly possible though to manage all of your products in one project too.

Keep in mind however that only one project can be loaded at the same time on a workstation. In case an operator needs to be able to select a particular product or product variant to assemble, these products or product variants should be managed within the same project.

3.2 Creating a project

What?

When you open the Arkite software for the first time, no projects will be available. The first thing you need to do is to create a project.

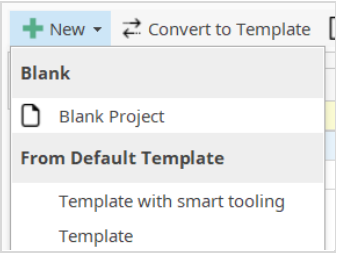


Steps

To create a new project, execute the following steps:

1. On the **Projects** tab and the **Projects** subtab, click the **New** button.

2. To create a project from scratch, select **Blank Project**.
   To create a project from a template, select a template under **From Default Template**.
   Two templates are available, one with and one without smart tooling. You can find more information on the contents of the templates in the next section.



3. In the **Properties** pane, update the project properties:
   - In the **Name** field, provide an appropriate name for your project

     Note

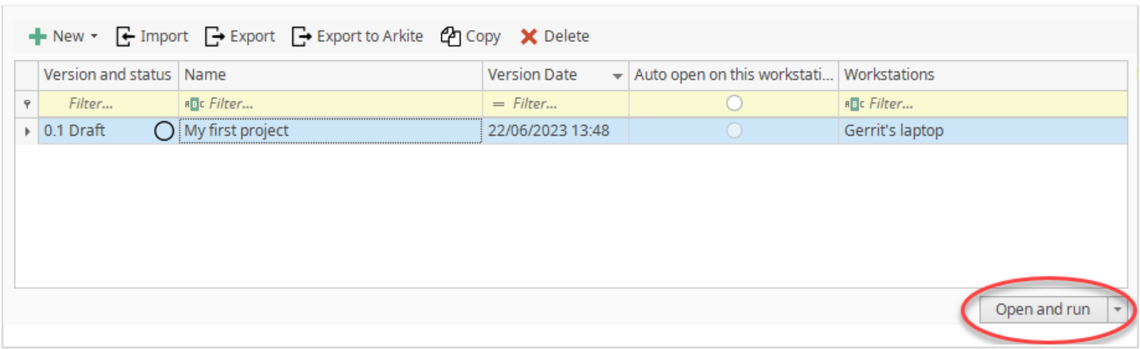
     *Next to the **Name** label, the object ID for the project is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
   - In the **Description** field, provide a short explanation of the purpose of the project.
   - If you want to provide more information, enter it in the **Details** field.
   - If you want to load the project automatically when starting the Arkite software, check the **Auto open on this workstation on startup** checkbox.

   Note

   - *The checkbox can only be checked for projects that have a production version, since projects that only have a **Draft** version are only available to Admin users.*
   - *The project status can be left in **Preparation** at creation. For more information about project statuses, refer to section **3.3 Managing projects** on page 34.*

Result

Your project is added to the project list and you are ready to go. To start configuring your project, click the project name or select the project and click **Open and run**.



Note

*In case of issues with your project, for example a communication loop, you can open your project without running it to troubleshoot. To do so, select the project, click the arrow on the **Open and run** button and select **Open without running**.*

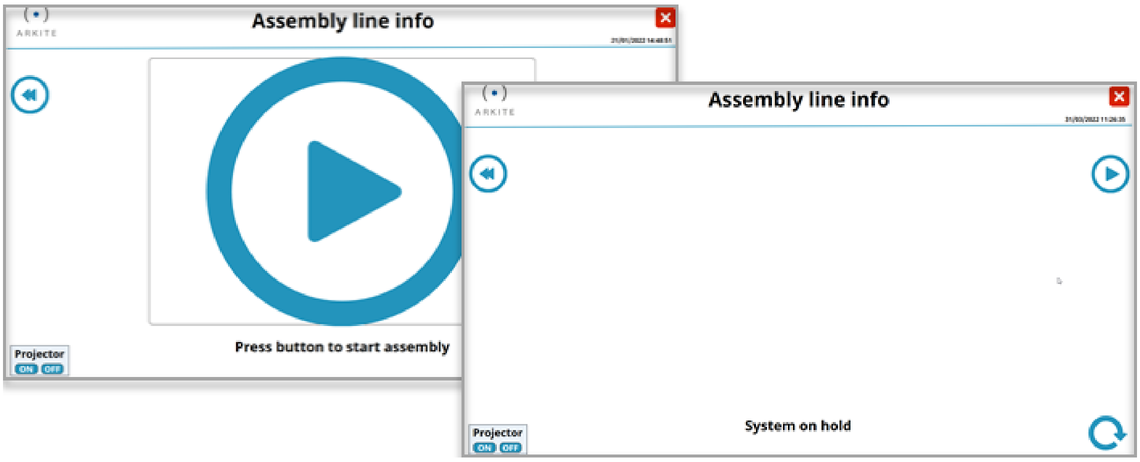
Default templates

Two default templates are provided with the Arkite software:

- With Smart Tool integration
- Without Smart Tool integration

Both templates contain a layout for your **Screen** display, containing:

- The Arkite logo
- The current date and time
- A button allowing you to close the **Screen** display
- Buttons to allow you to:
  - Reset your assembly process
  - Pause your assembly process
  - Go to the previous step or next step in your assembly process
- Buttons to turn your projector on or off



Furthermore, the templates contain the required logic to update the date and time, and a default process in which you can start building your assembly logic. For more information, refer to chapter **5 Processes** on page 96.

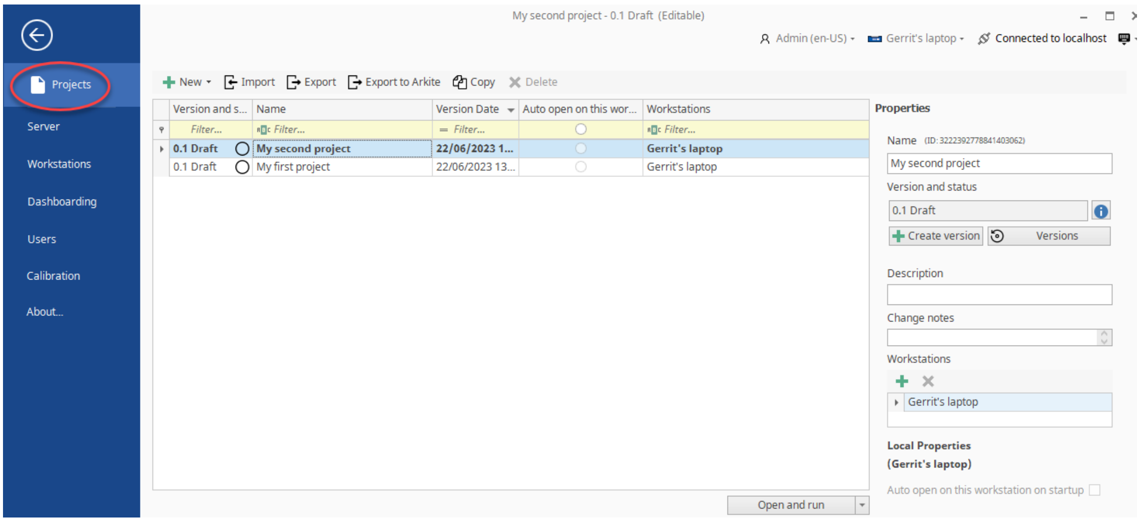
In the template with smart tool, the basic configuration for connecting to a smart tool through Open Protocol is also available:

- An Open Protocol communication module has been added. You just need to update the properties to match settings for your smart tool. For more information, refer to section **Open Protocol** on page 173.
- A Smart Tool detection has been added, with the related tool connection settings. You just need to teach it (when using **Validate** license) and link it to the proper activity detections. For more information, refer to sections **4.5 Teaching and finetuning detections (Validate and Vision license only)** on page 64 and **4.8 Working with smart tools** on page 83.

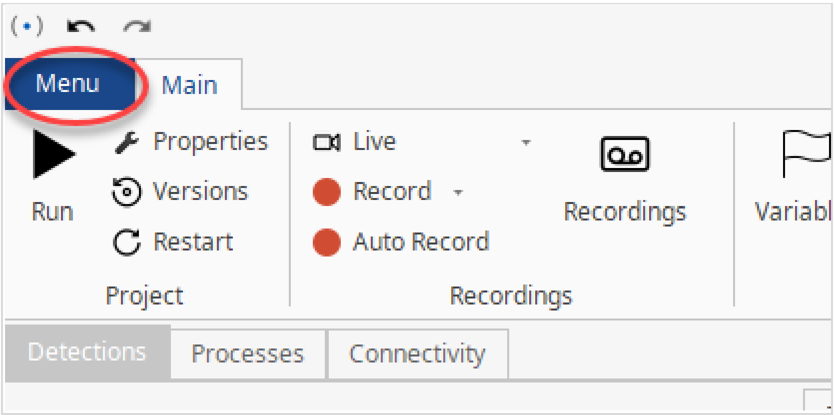
3.3 Managing projects

What?

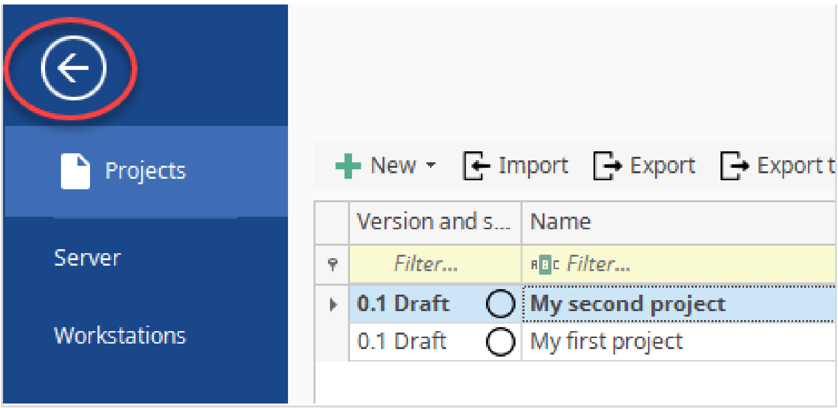
You can find an overview of all projects that are available on your workstation in the **Projects** menu. Only one project can be loaded on the workstation at the same time. That project is rendered in **bold**.



When you are working in a project, you can always return to your project overview by clicking on the **Menu** tab.



To return to your project, click the arrow in the upper left corner.



How to manage projects

On the **Projects** tab, you can:

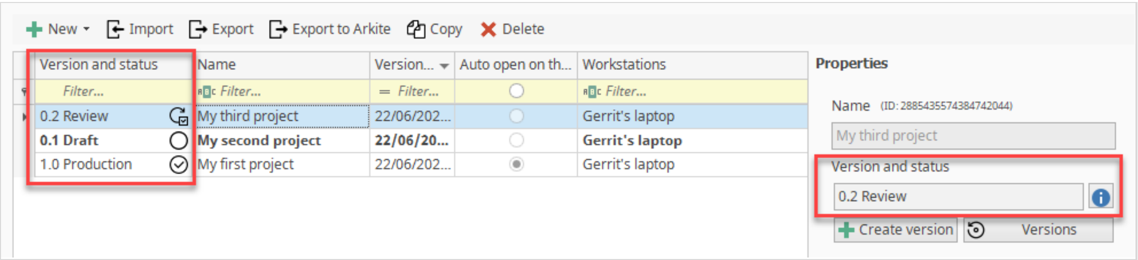
- Click **Import** to import a project prepared on another server
- Select a project and click **Export** to make it available for use on another server.
- Select a project and click **Export to Arkite** to provide a copy to Arkite for support.
- Select a project and click **Copy** to duplicate it.
- Select a project and click **Delete** to delete it.

3.4 Versioning

3.4.1 About versioning

What?

Every project has a specific version and status at all times. The current version and status of a project is shown in the **Version and status** column in the project overview as well as in the **Properties** pane.



Project versions

A project's version consists of a major and a minor version number.

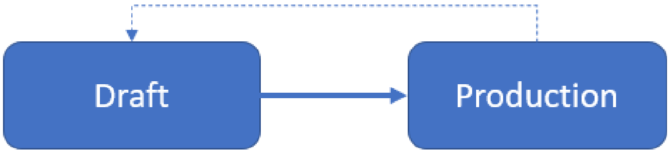
Every time a new version is created, the minor version number increases with one digit, e.g. from 0.1 to 0.2.

When the project status is changed to **Production**, the major version number increases with one digit and the minor version number is reset to 0, e.g. from 0.4 to 1.0.

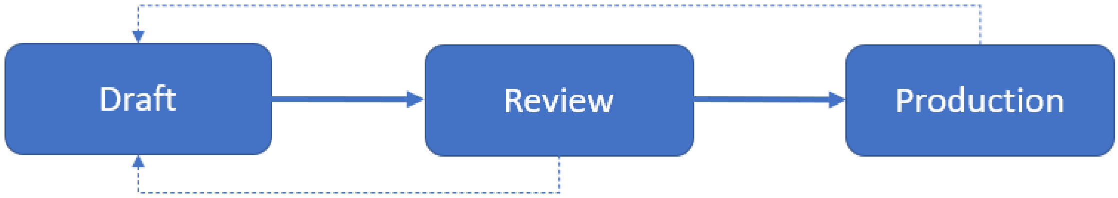
Project workflows

The Arkite software supports different project workflows:

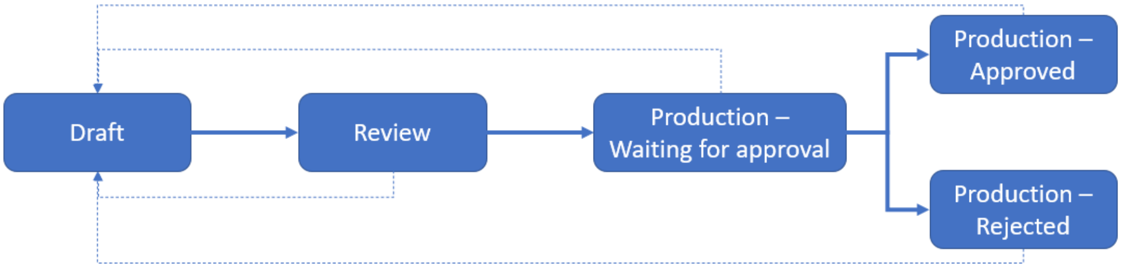
- The most simple workflow only discriminates between projects that are still being configured and projects that are being used in production. Once the project has been finalized, the project status is changed from **Draft** to **Production**.



- In a more advanced setup, a **Review** status is added to the workflow. In this status, the project is validated on all workstations before it is being put into **Production**.



- In the most advanced workflow, a project needs to be approved by one or more approvers before it can be used in production.



Note

*For more information on how to define which workflow is implemented, refer to section **3.4.2 Versioning settings** on page 37.*

Project statuses

A project can have the statuses in the table below. Depending on the project workflow that is implemented, one or more statuses can be skipped.

Icon	Status	Description
○	Draft	The project is being prepared offsite.
	Review	The project is being validated on all workstations.
	Production	The project is ready to be used in production.
	Production – Awaiting approval	The project is ready to be used in production, but still needs to be approved by one or more approvers.
	Production – Approved	The project is ready to be used in production and has been approved by all required approvers.
	Production – Rejected	The project was ready to be used in production but was rejected by at least one approver.

Note

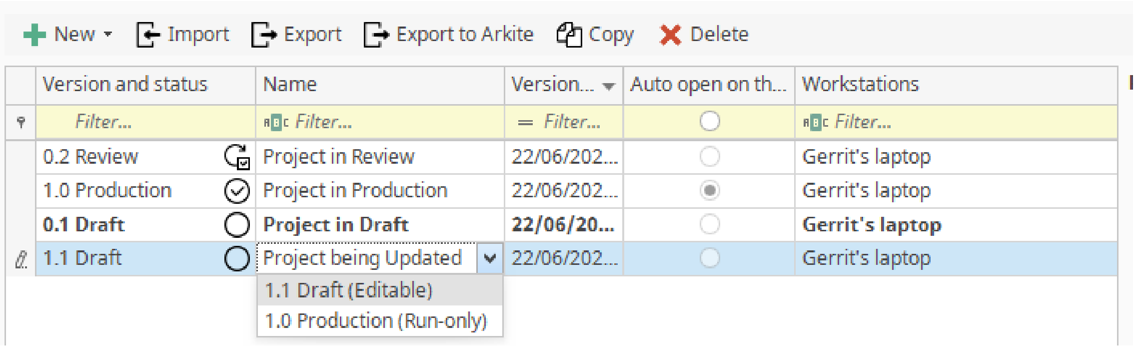
- *In the workflows in which the **Review** status is used, only projects with status **Draft** can be edited or removed.*
- *In the **Draft-Production** workflow, the **Admin** user can still edit a project in status **Production** as long as no new **Draft** version has been created.*

A project needs to be in status **Production – Approved** in order to be used in production. Operators (i.e. standard users without permissions to create or manage projects in the Arkite software) will only be able to open and run projects in status **Production – Approved**.

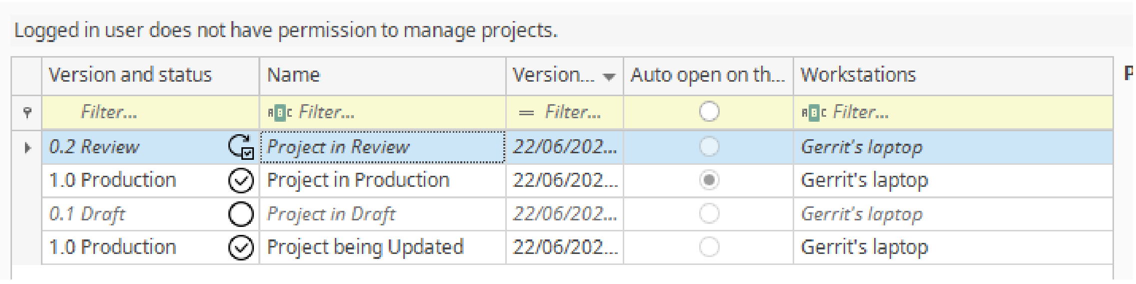
As an example, consider a case in which four projects are available:

- a project in status **Draft** (version 0.1)
- a project in status **Review** (version 0.2)
- a project in status **Production – Approved** (version 1.0)
- a project for which a production version exists, but is currently being updated (status **Draft**, version 1.1)

Users with admin permissions will be able to see and open all projects. For the project with version 1.1, he can choose to open the production version or the current draft version.



Standard users can only open the production version of a project. They won't see the draft version of the project that is being updated. The two projects that do not have production version yet are displayed in italics. They cannot be opened.



3.4.2 Versioning settings

What?

In the versioning settings, an admin user can define which project workflow is used.

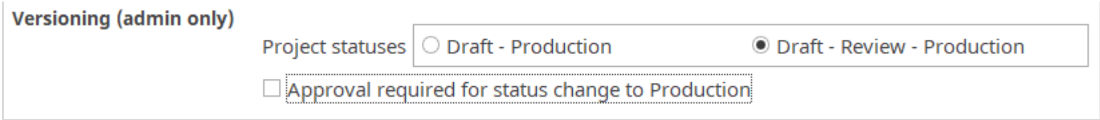
How to manage versioning settings

To manage the versioning settings, execute the steps below:

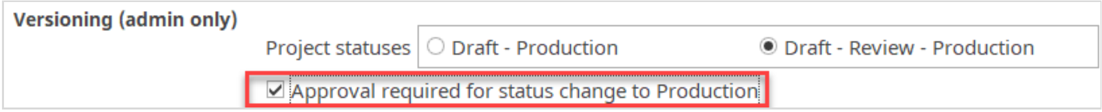
1. Click the **Menu** tab.
2. Click **Server** and then **Settings**.

3. Under **Versioning**, select the workflow you want to be used:
   - If you only want to use the statuses **Draft** and Production, select the **Draft – Production** radio button.

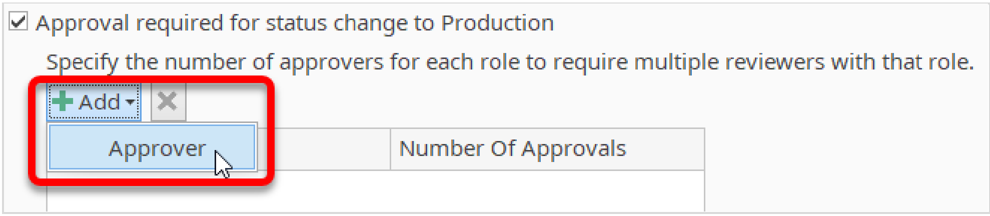
   - If you want to use the status **Review** too, select the **Draft – Review – Production** radio button.



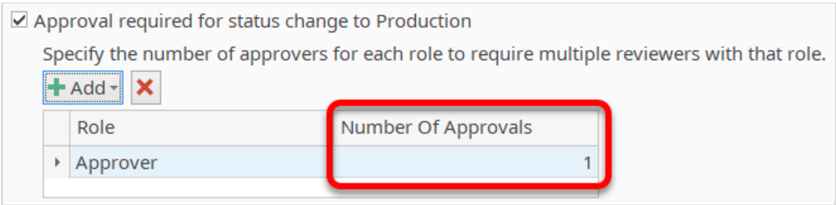
4. In case you want one or more users to approve a project before it is used in production:
   - Check the **Approval required for status change to Production** checkbox.



   - Click the **Add** button and select the role of the user that needs to approve the project.



   - To define multiple approvers with different roles, add all the required roles.
   - To define multiple approvers with the same role, change the number of approvers in the **Number of Approvals** column.



Note

*The role you add here should have the **Approve Projects** permission. By default, the **Approver** role has this permission.*

*For more information on managing roles and users, refer to chapter **16 User management** on page 241.*

3.4.3 Creating a new project version

What?

You can create a new project version and update its status:

- From the project overview
- From within a project

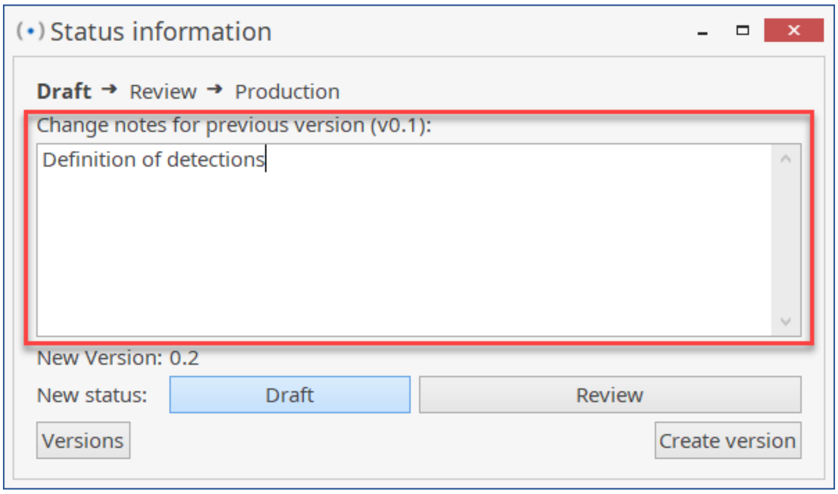
How to create a new project version from the project overview

To create a new project version from the project overview, execute the steps below:

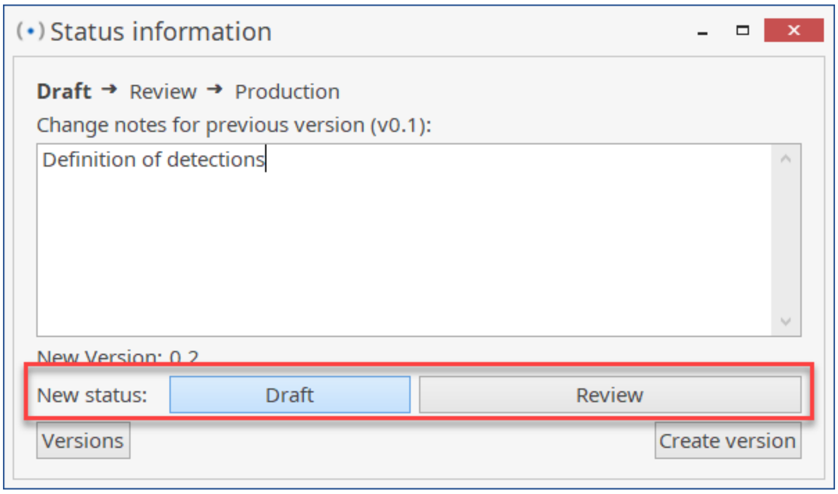
1. Click the **Menu** tab.
2. Select the project for which you want to create a new version and/or update the status.
3. In the **Properties** pane under **Version and status**, click **Create version**.

The **Status Information** window appears.

4. In the **Change notes for previous version** text field, describe which changes have been made to the project in the current version.

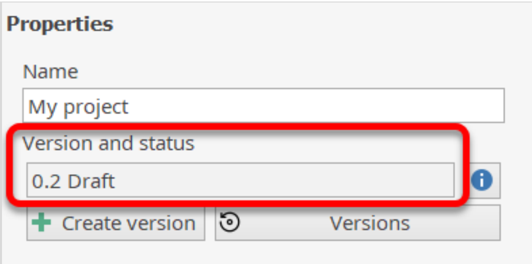


5. Under **New Status**, select the project's status.



Note
The statuses you can select here depend on the project's current status and on the project workflow that you have defined.

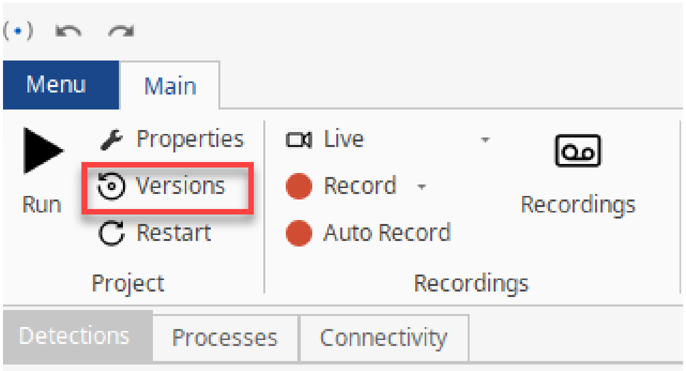
6. Click **Create Version**.
   A new version with a new version number is created.



How to create a new project version from within the project

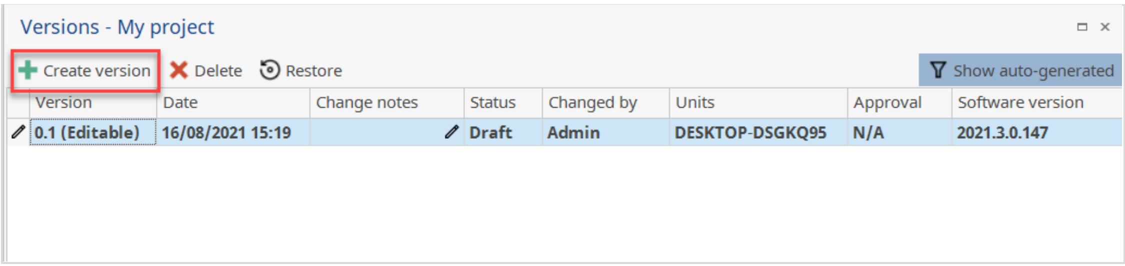
To create a new project version from within the project, execute the steps below:

1. In the ribbon, in the **Project** group, click **Versions**.



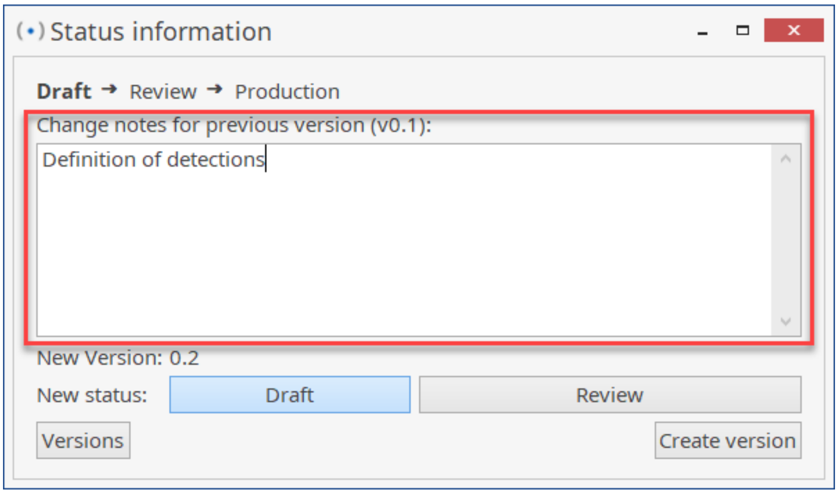
The **Versions** window appears.

2. Click **Create Version**.

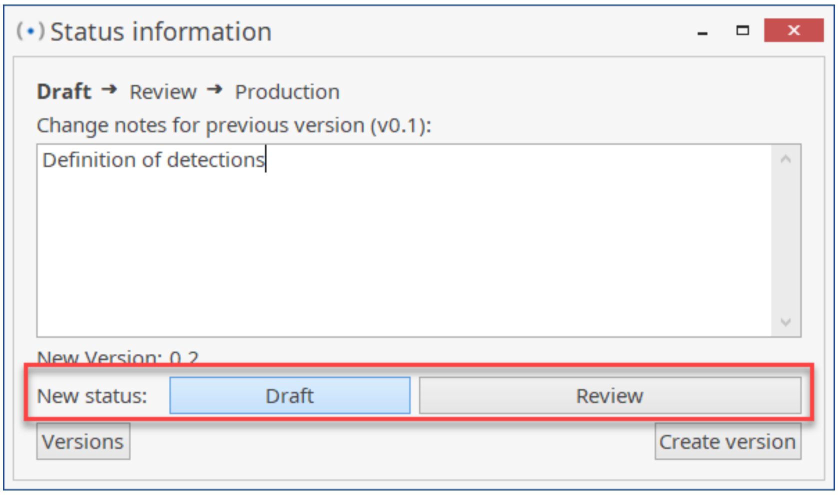


The **Status Information** window appears.

3. In the **Change notes for previous version** text field, describe which changes have been made to the project in the current version.



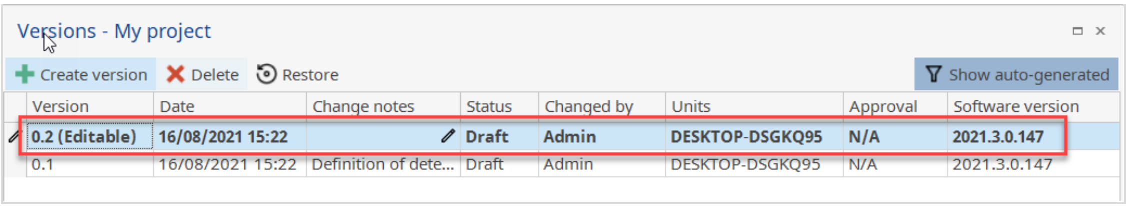
4. Under **New Status**, select the project's status.



Note

The statuses you can select here depend on the project's current status and on the project workflow that you have defined.

5. Click **Create Version**.
   A new version with a new version number is created.



3.4.4 Approving or rejecting a project

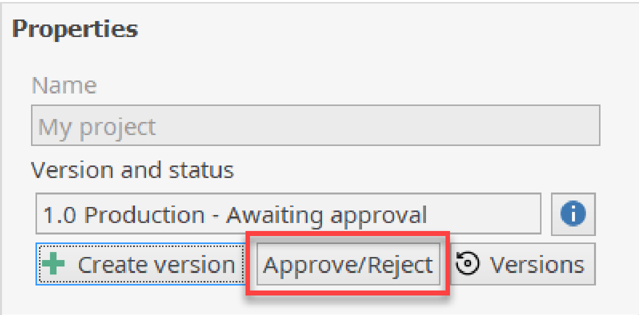
What?

In case the project workflow with approval is configured, one or more users with a role which has the **Approve Projects** permission need to log in to the Arkite software and approve the project, before it can be used in production.

How to approve a project

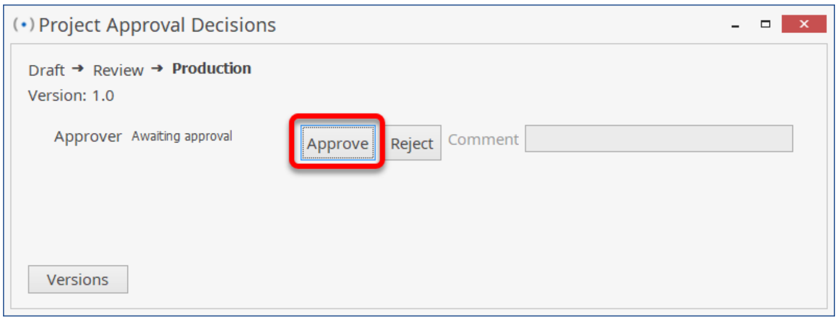
To approve a project, execute the steps below:

1. Log in to the Arkite software with a user that has a role with the **Approve Projects** permission. By default, the **Approver** role has this permission.
2. Select the project with status **Production – Waiting for Approval**.
3. In the **Properties** pane under **Version and status**, click **Approve / Reject**.



   The **Project Approval Decisions** window appears.

4. To approve the project, click **Approve**.



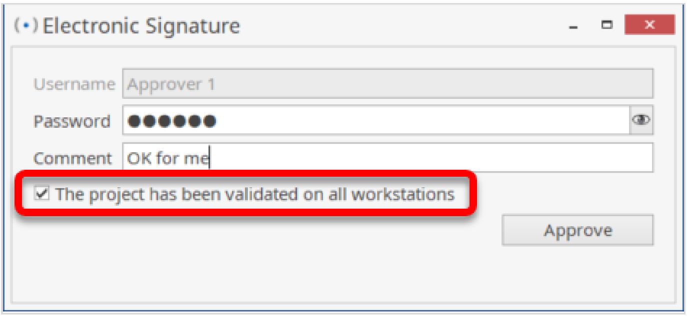
   The **Electronic Signature** window appears.

5. Enter your password and, optionally, a comment.

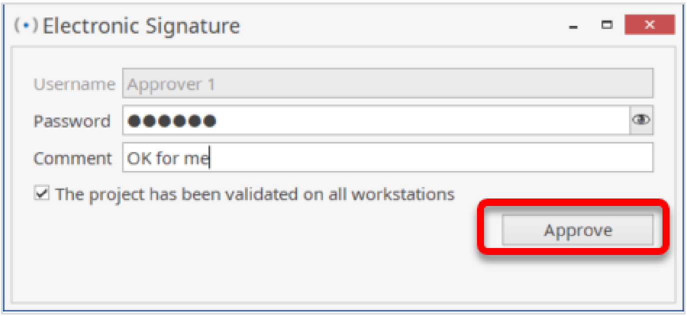
   Note

   You can click the button to make the password you've entered visible and verify if you haven't made any typing mistakes.

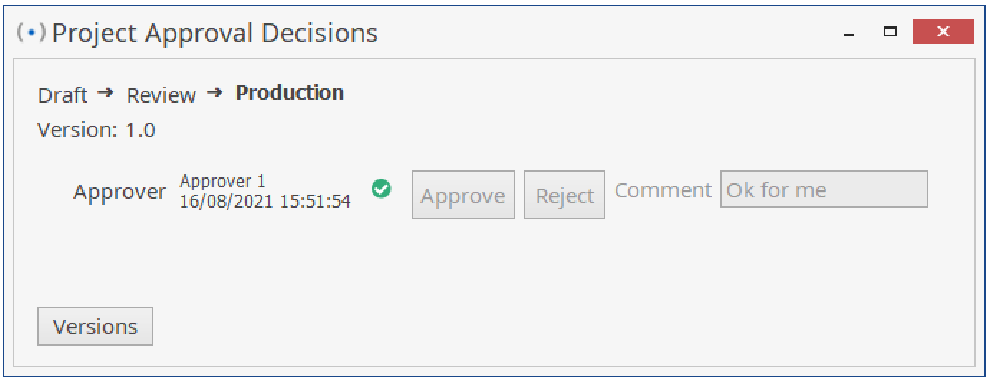
6. Check the **The project has been validated on all workstations** checkbox.



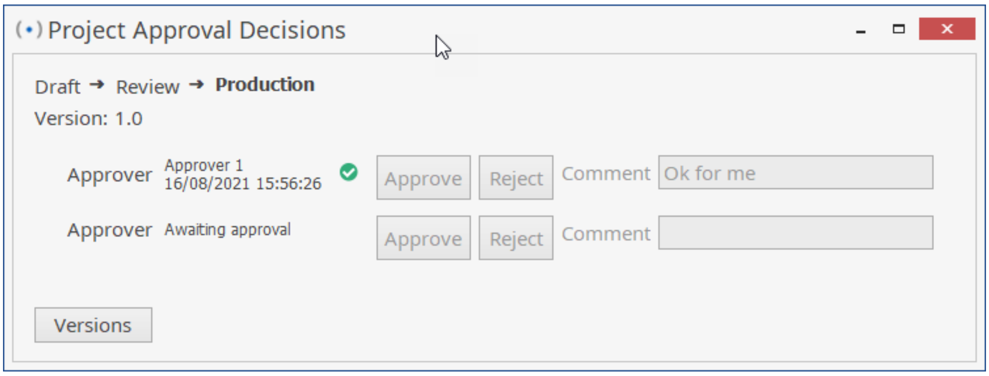
7. Click the **Approve** button.



The approval is registered.
If you were the only or the last approver, the status is changed to **Production – Approved**.



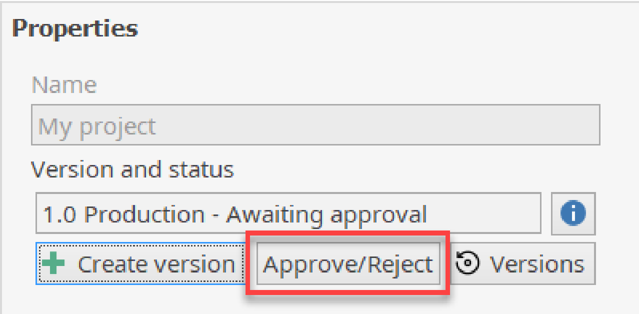
If more approvals are required, the status remains in **Production – Waiting for Approval**.



How to reject a project

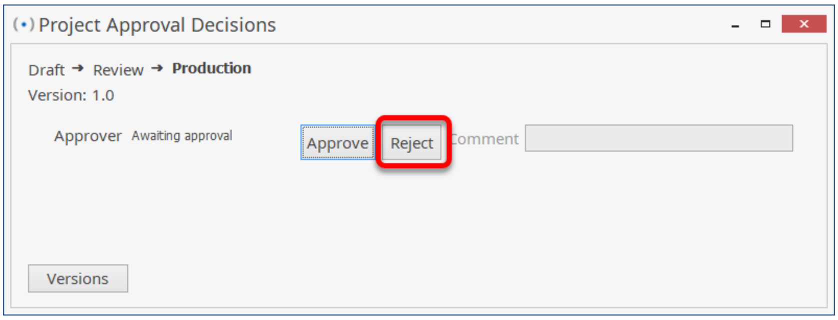
To reject a project, execute the steps below:

1. Log in to the Arkite software with a user that has a role with the **Approve Projects** permission. By default, the **Approver** role has this permission.
2. Select the project with status **Production – Waiting for Approval**.
3. In the **Properties** pane under **Version and status**, click **Approve / Reject**.



The **Project Approval Decisions** window appears.

4. To reject the project, click **Reject**.



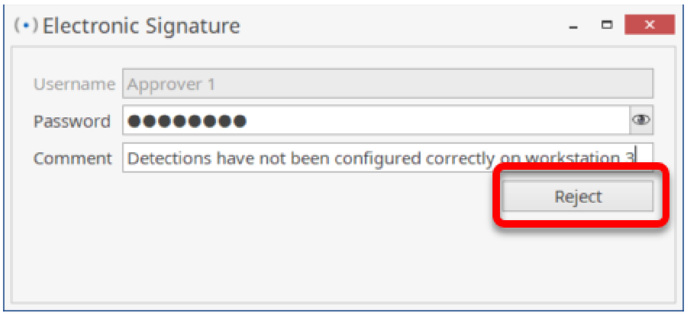
The **Electronic Signature** window appears.

5. Enter your password and a reason for the rejection in the **Comment** field.

Note

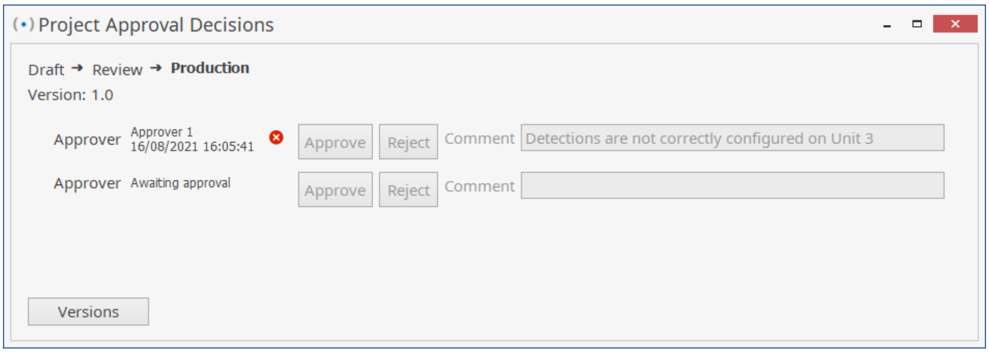
You can click the button to make the password you've entered visible and verify if you haven't made any typing mistakes.

6. Click the **Reject** button.



The rejection is registered.
The status is changed to **Production – Rejected**.

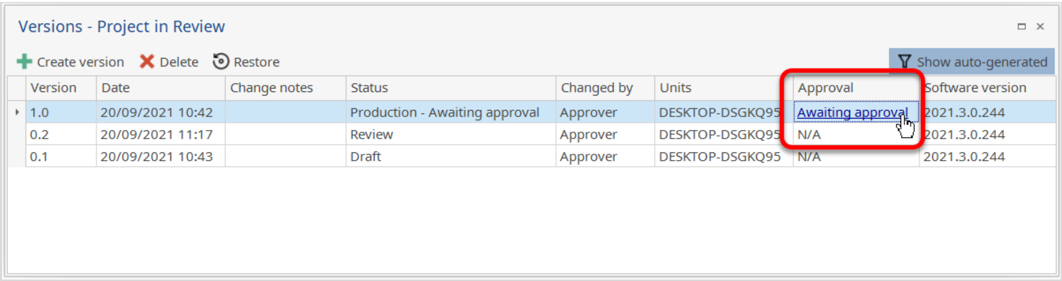
Any additional approvers will no longer be able to approve the project at this stage.



Note

It is also possible to approve or reject a project from the project's version history:

- *Select the project you want to approve or reject.*
- *In the **Properties** pane under **Status**, click **Versions**.*
- *In the **Approval** column, click the text **Awaiting approval**.*
- *Approve or reject the project as explained in the section above.*



3.4.5 Opening the latest production version

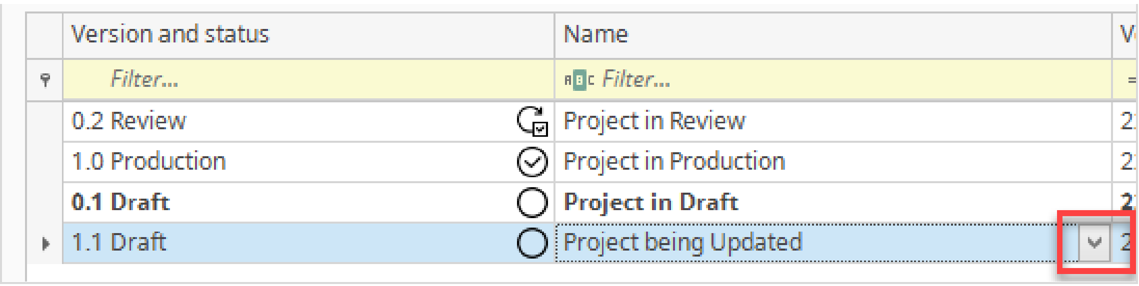
What

Users with the editor or approver role can choose to open the latest approved Production version or the current version of a project. This way, other workstations can still use the latest production version of the project even though it is being edited.

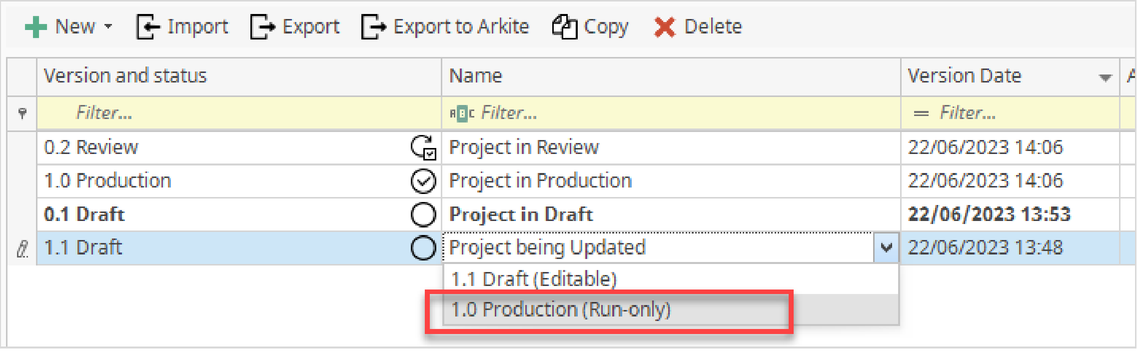
Steps

To open the latest production version of a project, execute the following steps:

1. On the **Projects** tab, select a project with at least one production version, i.e. the version should at least be 1.1.
2. In the **Name** column, click the arrow at the right side of the cell.



3. From the dropdown list, select the latest production version (in this case version 1.0).



Result

The latest production version opens in run-only mode.

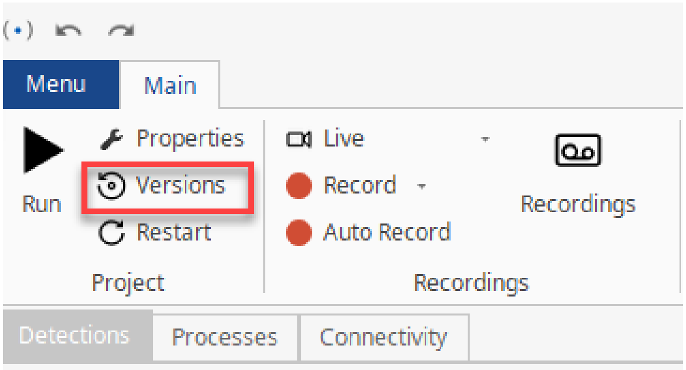
3.4.6 Version history

What?

The Arkite software keeps a history of all previous versions of the project. You can consult it and restore a previous version if required.

How to consult the version history

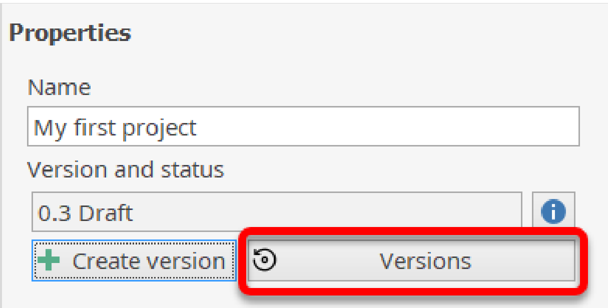
To consult the version history, click **Versions** in the **Project** group on the ribbon.



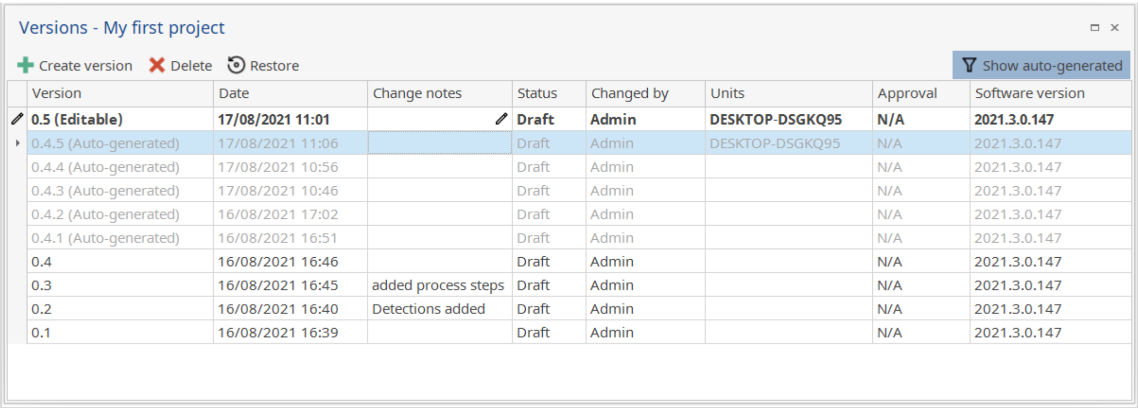
Note

You can also open the version history from the project overview:

- *Select the project for which you want to see the version history.*
- *In the **Properties** pane under **Version and status**, click **Versions**.*



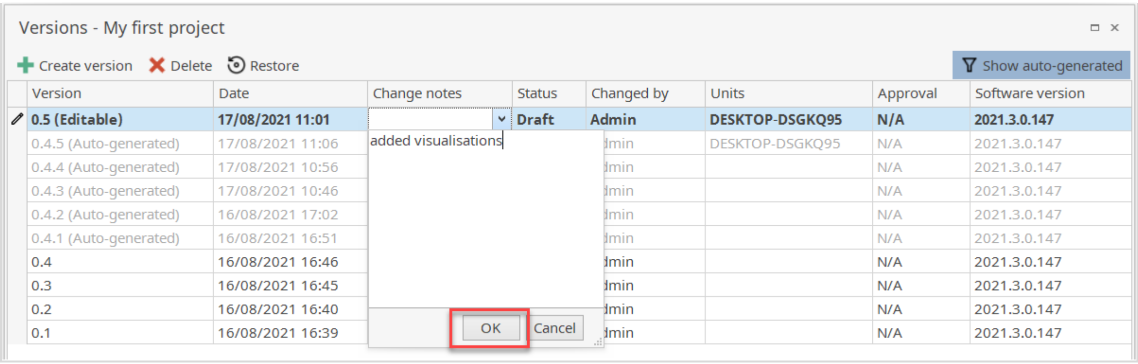
In the **Versions** screen, you get an overview of all previous versions of a project, with their timestamp, change notes and status.



To read the complete change notes for a version, click the corresponding cell in the **Change Notes** column.

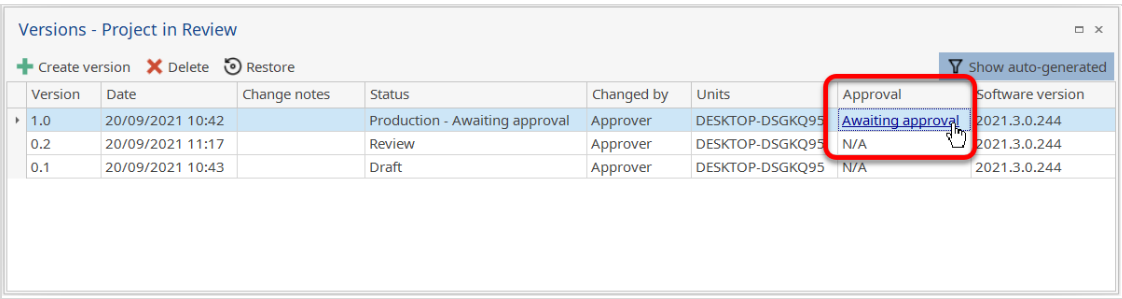
For the current version, you can edit the change notes:

- Click the cell in the **Change Notes** column
- Enter your text
- Click **OK**.



Projects in status **Production – Awaiting Approval** can also be approved or rejected from here, provided the user that is logged in has the required role:

- In the **Approval** column, click the text **Awaiting approval**.
- Approve or reject the project as explained in section **3.4.4 Approving or rejecting a project** on page 42.

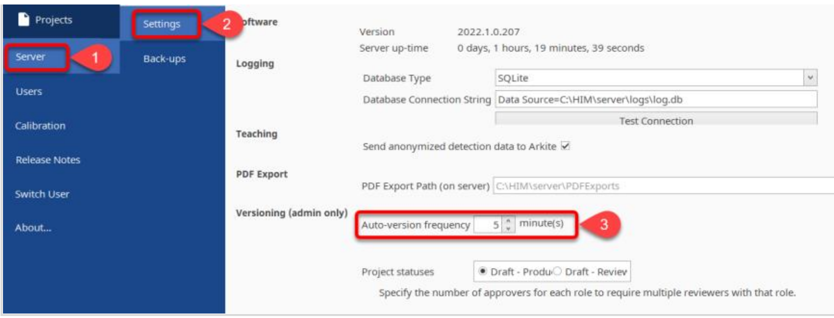


While you are working on your project, the Arkite software automatically generates a version every five minutes (provided there have been any changes). This way, you can always return to a very recent version of your project when something goes wrong.

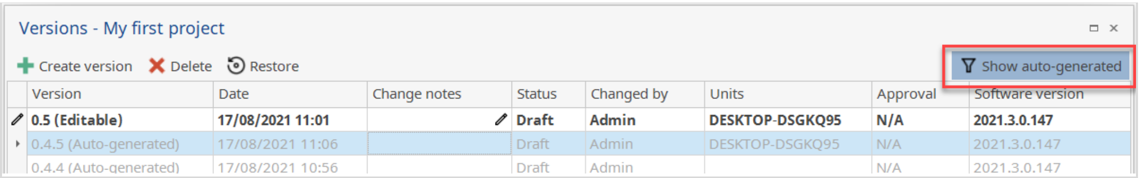
Note

You can configure the frequency with which the Arkite software auto-generates a version. To do so:

- *Click the **Menu** tab.*
- *Click **Server** and then **Settings**.*
- *Under **Versioning (admin only)**, update the value in the **Auto-version frequency** field.*

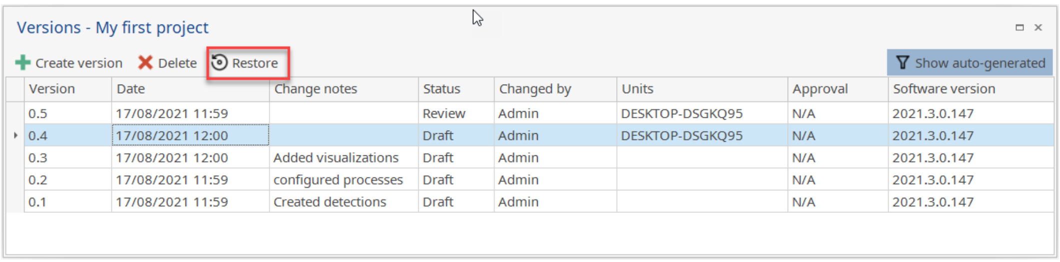


To include the versions that have been automatically generated, click the **Show Auto-generated** button.

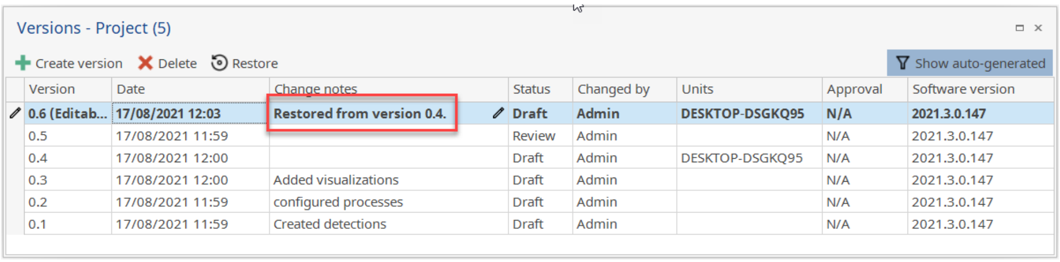


How to restore a previous version

To go back to a previous version, select the version and click **Restore**.



A new draft version of the project is created. In the change notes, it's indicated from which version the restore has been made.



How to delete a previous version

To delete a previous version select the version and click **Delete**.

> ***Note***
>
> *If the workflow with required approval is configured, it is not possible to delete any of the previous versions of a project.*

3.5 Working on projects with multiple authors

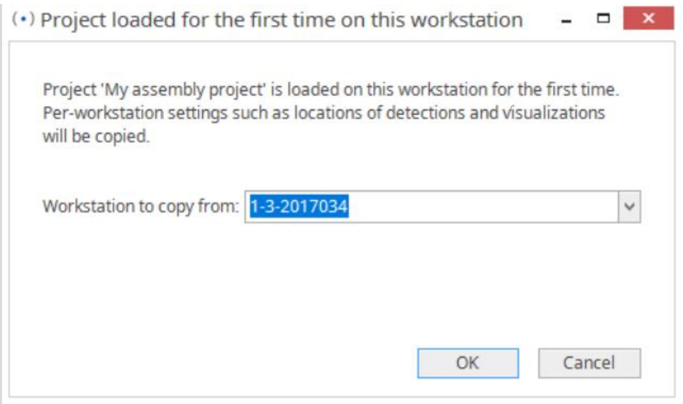
What?

In your project overview, you have access to all projects on any workstation connected to the same server. This also allows you to work on the same project with multiple authors, and to use the same project on multiple workstations. If for example the same or a very similar assembly process is taking place on several workstations, the project will be stored on the server and can be edited and maintained from different workstation.

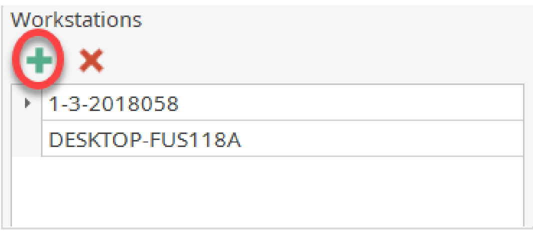
Opening a project created on another workstation

In your **Projects** overview on the **Menu** tab, you will see all projects that are on the server, also the ones that were created on another workstation.

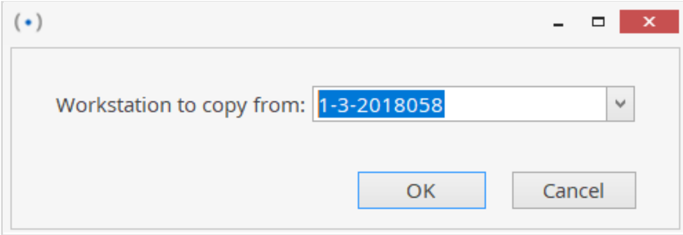
When opening a project for the first time on your workstation, the complete project is copied from another workstation, including all workstation specific settings and properties such as visualizations and the location of detections. The popup below will appear when opening the workstation. Select the workstation you want to copy from and click **OK**.



> ***Note***
>
> *If you want to open the project on an editor, i.e. a workstation without a license, the system will by default switch to another workstation. If you want to open the project on the editor, you first need to add the editor to the workstations in the project properties:*
>
> - *Select the project*
> - *In the **Properties** pane under **Workstations**, click the + button.*



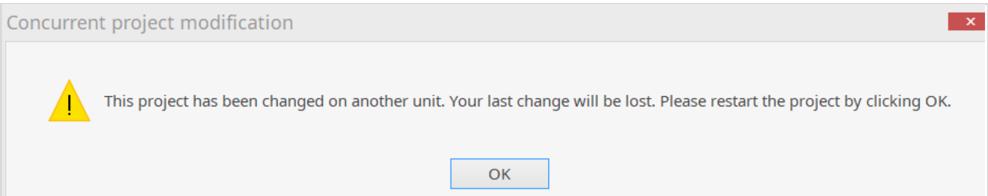
- *Select a workstation to copy from and click **OK**.*



Editing a project created on another workstation

From that moment, you can start editing the project yourself. Anything you add or change, will also be immediately available on the other workstation.

> ***Note***
>
> *If the project is being edited on several workstations simultaneously, it's possible the project is changed on another workstation before your change can be applied. In that case, the pop up below will appear. Your change will be reverted, the project will be restarted.*



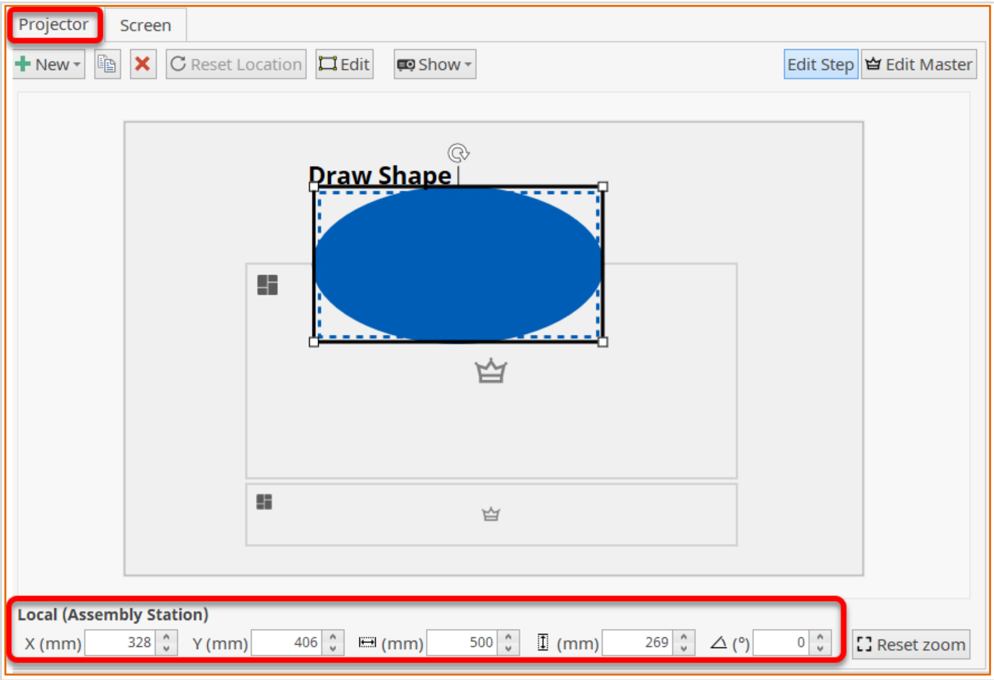
Global properties versus workstation-specific properties

Some properties and settings are **global**. This means they are valid for the project as such and are always exactly the same on each workstation. This applies for example for detections. If you add a detection, change its name or delete it, this will have an immediate effect on all other workstations.

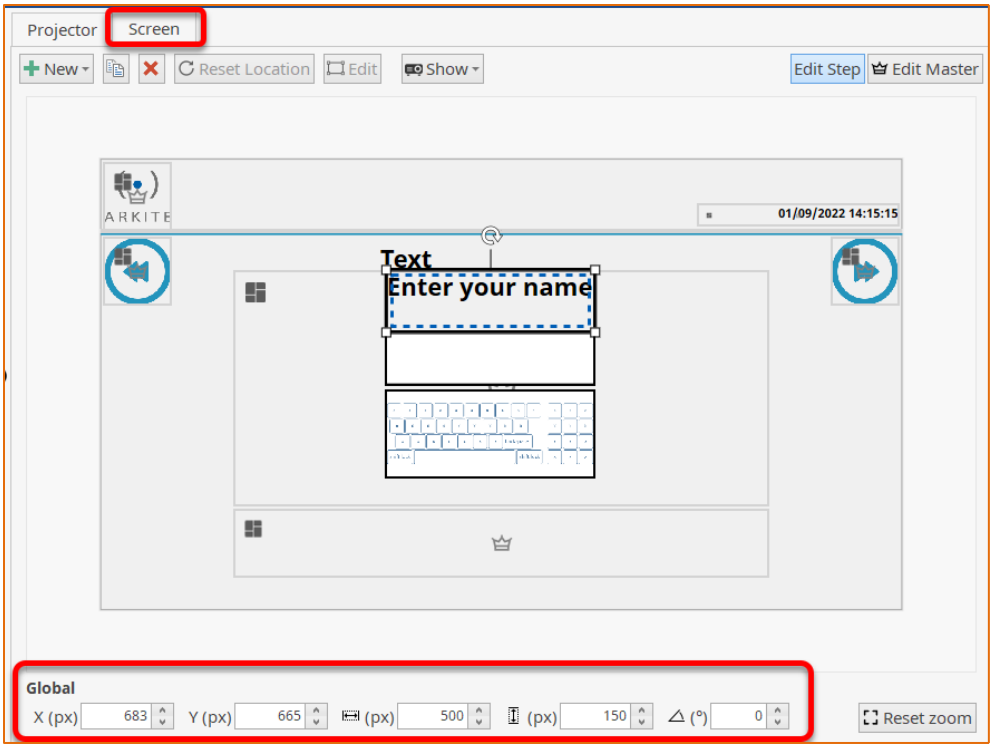
Some properties and settings are **workstation specific**. This means they can be different on every workstation. This applies for example for the **location** of a detection. If you add a detection and teach it, the detection and its location is copied on all other workstations. However, if you change the location afterwards, e.g. by reteaching it, the location will only change on that particular workstation.

In the user interface, it's always indicated if something is global or local/workstation specific.

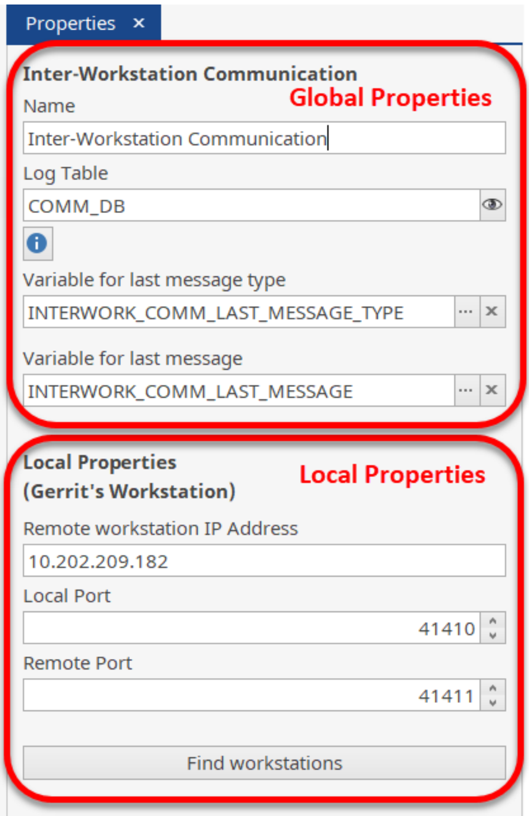
For example, the location of visualizations on a **Projector** display is local or workstation specific, as you can see on the screenshot below.



The location of visualizations on a **Screen** display is global however, as you can see on the screenshot below.



Some elements have properties that are global and others that are local. In the screenshot below, you can see this for an **Inter-Workstation** communication module:



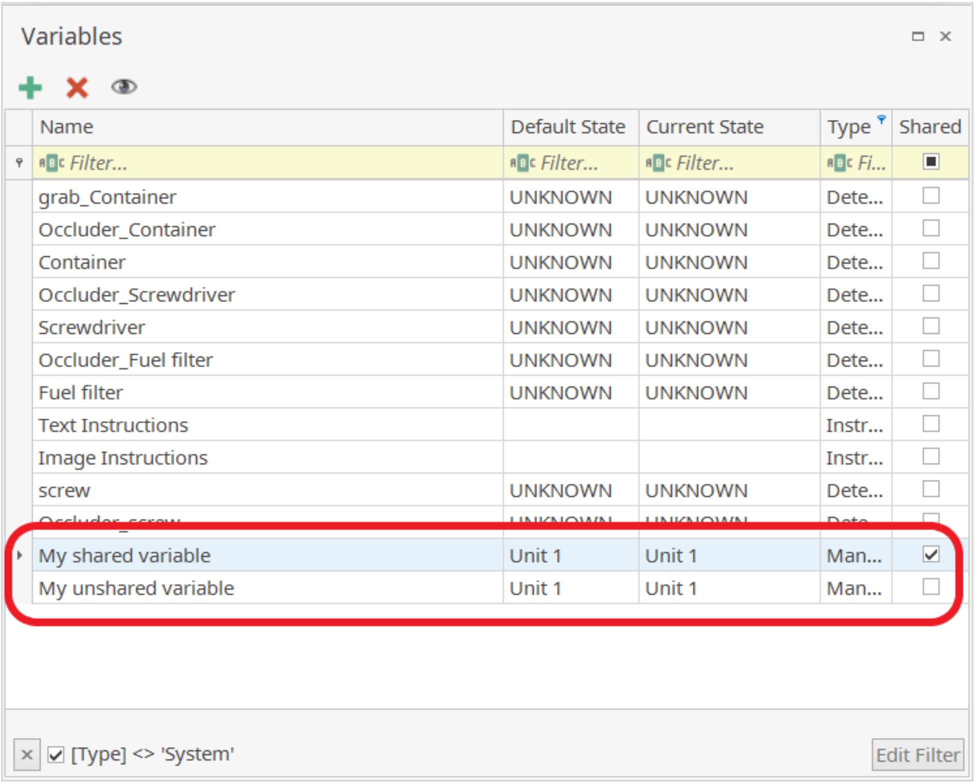
The table below gives an overview of the most important global and workstation specific aspects of a project:

Global	Local / Workstation specific
Project name, description and status	Location of a detection (**Validate** license only)
Detections	Location of visualizations on a **Projector** display
Processes	Display Port
Process steps	Value of a non-shared variable
Text and image instructions	
Displays	
Location of visualizations on a **Screen** display	
Variables	
Value of a shared variable	

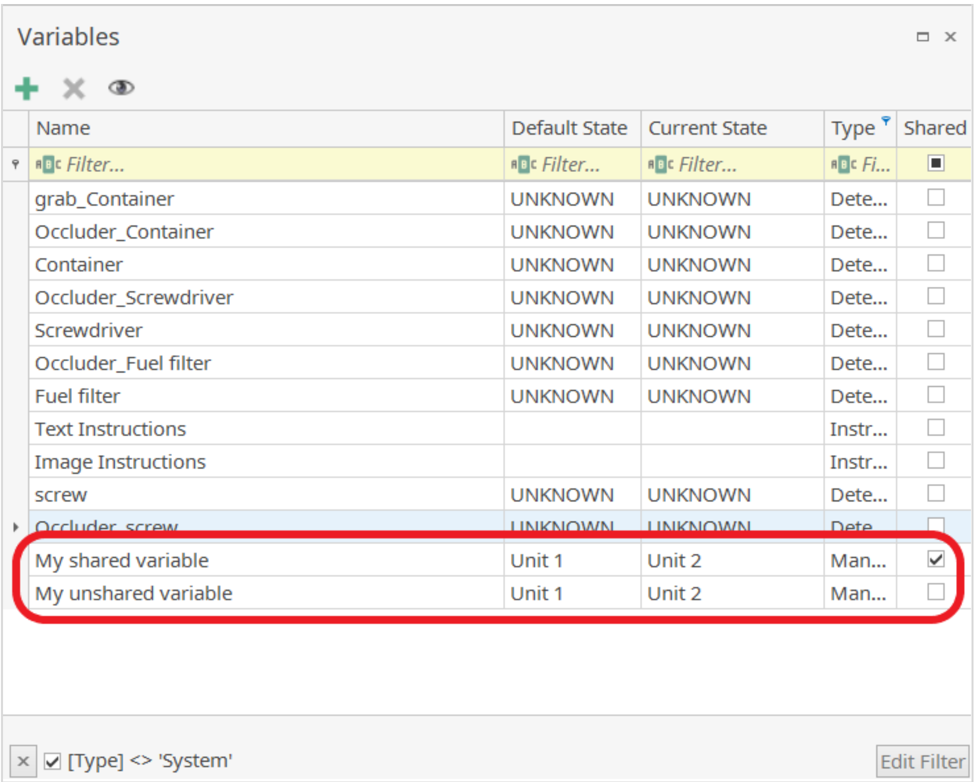
Sharing variables

For variables, you have the choice to synchronize the value across different workstations or to manage them independently. In order to synchronize values for a variable, check the **Shared** checkbox in the **Variables** screen.

In the example below, two variables have been created, one shared variable and one unshared variable. They both have the value **Workstation 1**. After creation, the variables are available on all workstations, with their corresponding values.



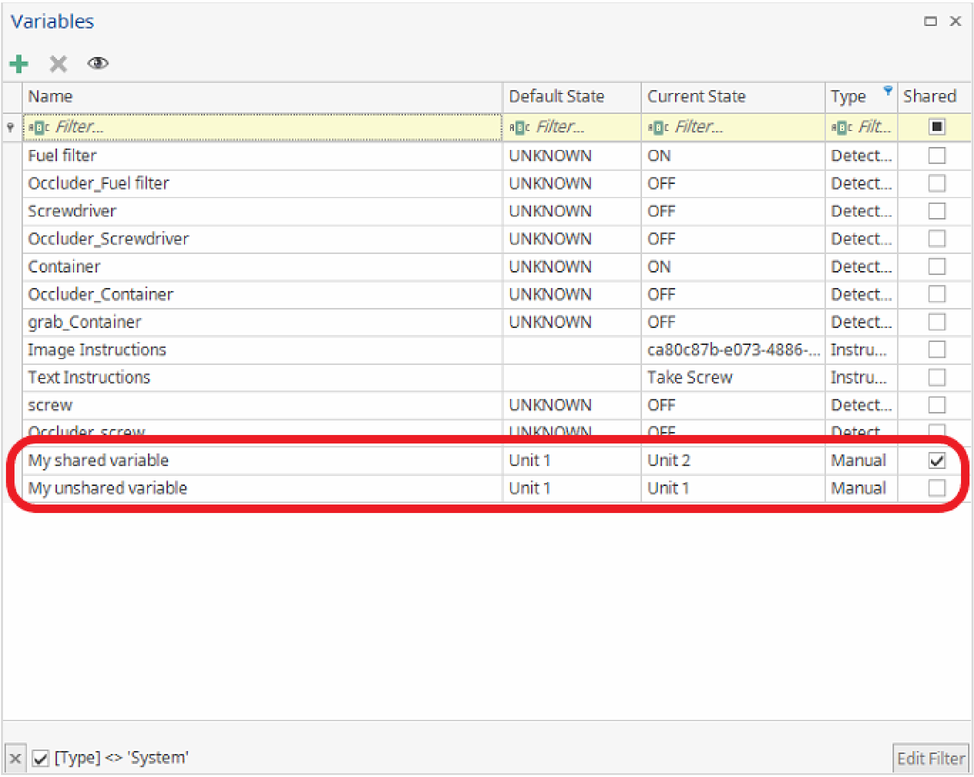
Then, the value of both variables is updated to **Workstation 2** on one of the workstations.


Variables

Name	Default State	Current State	Type	Shared
Filter...	Filter...	Filter...	Fi...	■
grab_Container	UNKNOWN	UNKNOWN	Dete...	☐
Occluder_Container	UNKNOWN	UNKNOWN	Dete...	☐
Container	UNKNOWN	UNKNOWN	Dete...	☐
Occluder_Screwdriver	UNKNOWN	UNKNOWN	Dete...	☐
Screwdriver	UNKNOWN	UNKNOWN	Dete...	☐
Occluder_Fuel filter	UNKNOWN	UNKNOWN	Dete...	☐
Fuel filter	UNKNOWN	UNKNOWN	Dete...	☐
Text Instructions			Instr...	☐
Image Instructions			Instr...	☐
screw	UNKNOWN	UNKNOWN	Dete...	☐
Occluder_screw	UNKNOWN	UNKNOWN	Dete...	☐
My shared variable	Unit 1	Unit 2	Man...	☑
My unshared variable	Unit 1	Unit 2	Man...	☐

☑ [Type] <> 'System' Edit Filter

On the other workstations, the value of the shared variable is updated to **Workstation 2** as well. The value of the unshared variable hasn't changed however: it is still **Workstation 1**.


Variables

Name	Default State	Current State	Type	Shared
Filter...	Filter...	Filter...	Filt...	■
Fuel filter	UNKNOWN	ON	Detect...	☐
Occluder_Fuel filter	UNKNOWN	OFF	Detect...	☐
Screwdriver	UNKNOWN	OFF	Detect...	☐
Occluder_Screwdriver	UNKNOWN	OFF	Detect...	☐
Container	UNKNOWN	ON	Detect...	☐
Occluder_Container	UNKNOWN	OFF	Detect...	☐
grab_Container	UNKNOWN	OFF	Detect...	☐
Image Instructions		ca80c87b-e073-4886-...	Instru...	☐
Text Instructions		Take Screw	Instru...	☐
screw	UNKNOWN	OFF	Detect...	☐
Occluder_screw	UNKNOWN	OFF	Detect...	☐
My shared variable	Unit 1	Unit 2	Manual	☑
My unshared variable	Unit 1	Unit 1	Manual	☐

☑ [Type] <> 'System' Edit Filter

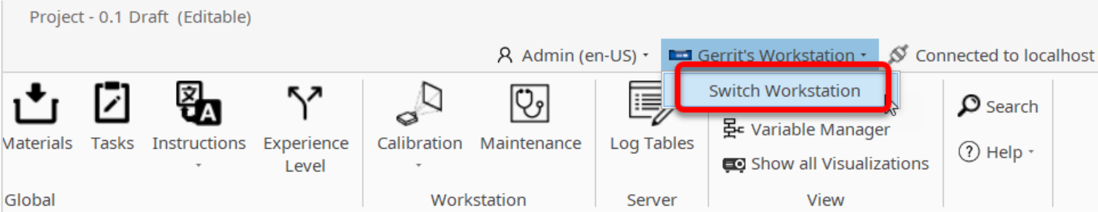
3.6 Switching workstations

What?

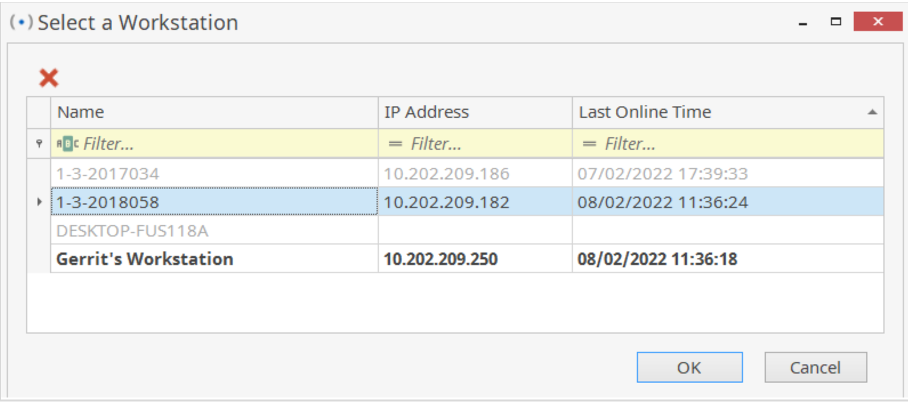
When you are working on an editor, i.e. a workstation that has not been assigned to a license, you can switch to another workstation on the same server. From then, your editor will ‘impersonate’ that workstation: all workstation specific properties from that workstation will be loaded, and all local properties you edit will be changed on that workstation and that workstation only. Also the license of that workstation will be used.

How to switch workstations

In the top bar, click your workstation name and select **Switch Workstation**.



In the **Select a Workstation** window, select the workstation of your choice and click **OK**.



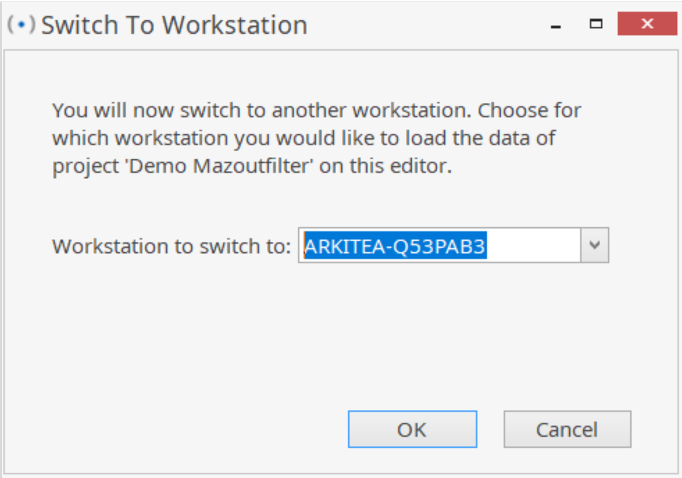
Note

If a project is open when you are switching workstations, the project will be closed after switching. You need to re-open it.

Automatic switch of workstation when opening a project

When you are working on an editor and open a project that has never been opened on that editor before, the system will automatically switch to another workstation.

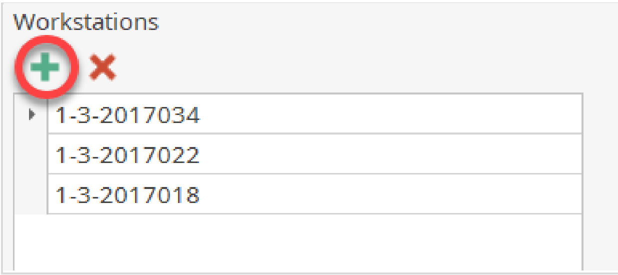
The **Switch to Workstation** pop-up will appear. Select the workstation you want to switch to and click **OK**.



Note

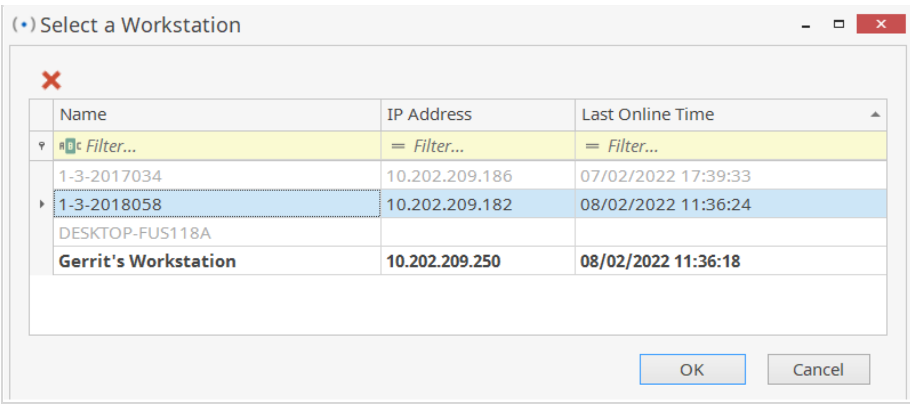
If you want to open the project on your editor instead of switching to another workstation, you first need to add the editor to the workstations in the project properties:

- *Select the project*
- *In the **Properties** pane under **Workstations**, click the* *button.*

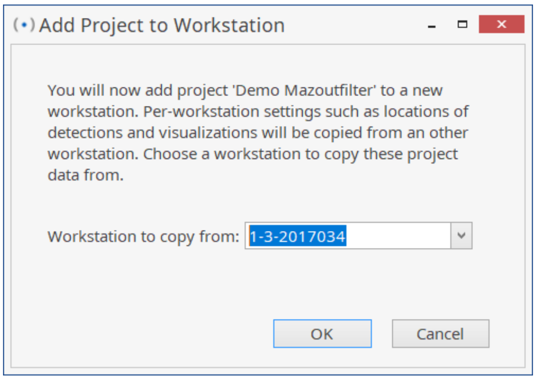


- *In the **Select a Workstation** window, select the workstation you want to add to the project and click **OK**.*

  By default, your own workstation is selected.



- *Select a workstation to copy from and click **OK**.*



4. Detections

4.1 About detections

A detection is a building block in the Arkite software used to make a representation of your workbench. By creating detections, you define which tools and components are used during and what activities take place during your assembly process. Once the detections have been configured in the Arkite software, you can use them to build your assembly process. This allows the system to guide the operator and give him the required instructions.

When using Arkite's operator guidance platform **with Arkite 3D sensor** (**Validate** and **Vision** license), it can also verify certain events or object states. The system will be able to tell your operator whether all required tools and objects are available and what to do with them. It can validate if an activity has been executed correctly and if a tool has been returned to its location.

Note

For more information about building your assembly process, refer to chapter ***5 Processes*** *on page 96.*

4.2 Detection types

In the table below, you can find a list of available detection types:

Detection type	Description
Object	Any component or product on the workbench. When using a **Validate** or **Vision** license (= with Arkite 3D sensor), the system will detect if the object is present or not and can give instructions to place it or take it away.
Tool	A tool that is used during the assembly process. When not used, the tool should be at a fixed location. A tool can also have a shadow board: this is an image of the tool that will be shown at the default location of the tool when it's not present. In case of a smart tool, the system can ensure that the tool can only be used during the correct step in the assembly process. When using a **Validate** or **Vision** license (= with Arkite 3D sensor), the system will detect if a tool is present or not and can give instructions to place it or take it away. For smart tools, it will not only ensure the tool is enabled at the right moment during the assembly process, but also at the right location. When the operator wants to use the tool at a wrong place, the tool will not work. When using a **Vision** license (= with Arkite Vision sensor), you can indicate a zone on your workbench in which a tool may be placed. The software will detect where exactly in this zone the tool is located and track it when you put it at a different location within the zone after usage.
Container	A storage unit or picking location, containing materials used in the assembly process. When using a **Validate** or **Vision** license (= with Arkite 3D sensor), the system will detect if the object is present or not and can give instructions to place it or take it away. Additionally, it can also ask to pick a material from the container and verify if the action is executed.
Activity	An activity that is performed during the assembly process. When using a **Validate** or **Vision** license (= with Arkite 3D sensor), the system can verify if an activity has been performed by detecting movement at the defined location. An activity can be linked with a tool.

Detection type	Description
Virtual button (system with Arkite 3D sensor only)	Only available when using a **Validate** or **Vision** license (= with Arkite 3D sensor)**.** A button that is projected on the workbench. The system can request an operator to press the button in order to make a choice or confirm an action.
Quality Check (system with Arkite Vision sensor only)	Only available when using a **Vision** license (= with Arkite Vision sensor). The expected state of an assembly process at a certain point in time. The system verifies if a (sub)assembly has been executed correctly by comparing the state of the workbench before the assembly step with the expected result after the assembly and, optionally, with images of results that should be considered wrong.

4.3 Detection states

Note

*Some detections states are only available when using a **Validate** or **Vision** license (= with Arkite 3D sensor). These are marked with a note **(Validate and Vision license only)** in the table below.*

A detection can have the states below:

State icon	Applicable for	Description
	all detections (Validate and Vision license only)	The detection is currently present on the workbench.
	all detections (Validate and Vision license only)	The detection is currently absent on the workbench.
	all detections (Validate and Vision license only)	The system cannot confirm whether the detection is currently present on the workbench, because something is blocking the sensor's view. Based on the latest state, the system assumes the detection is present.
	all detections (Validate and Vision license only)	The system cannot confirm whether the detection is currently present on the workbench, because something is blocking the sensor's view. Based on the latest state, the system assumes the detection is absent.
	Container (Validate and Vision license only)	Material is being picked from the container.
	all detections except for virtual buttons (Validate and Vision license only)	The detection state is not currently not calculated. This is the case for: – Activities and Quality Checks that are not active (**Run** mode) or not selected (**Edit** mode), unless the **Show Error Visualizations** property of the detection is checked. – Detections that are disabled due to the state of the variable in the detection's **Enabled** property.
	Container	A material has been associated with the container. If an image is available for the material, the icon will be replaced with this image.
	Container	No material has been associated with the container.

State icon	Applicable for	Description
	Activity	The activity is linked with a tool.
	Tool	The tool has been configured as a smart tool.

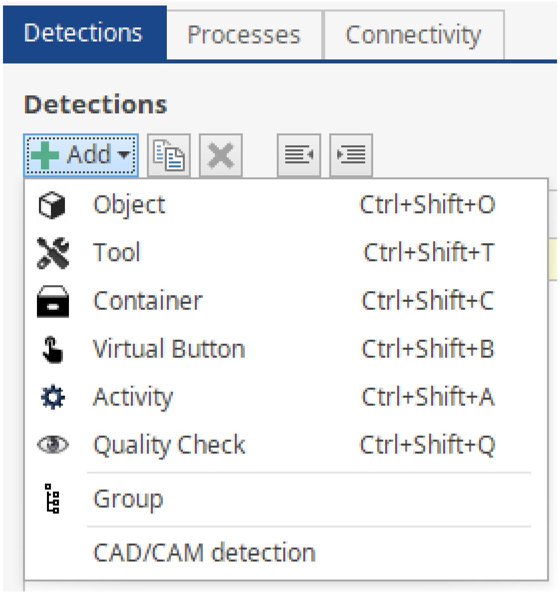
Note

*When using a **Validate** or **Vision** license (= with Arkite 3D sensor), detections that have been added but have not been taught to the system yet will not have any state.*

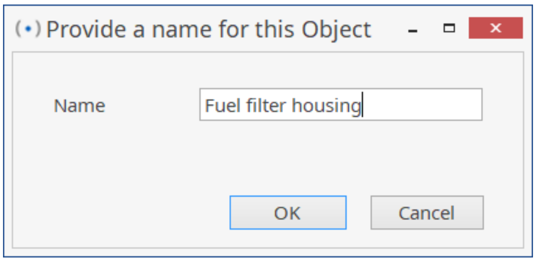
4.4 Adding detections

To add a new detection, execute the steps below:

4. Navigate to the **Detections** tab.
5. Click the **Add** button and select the type of detection you want to add.

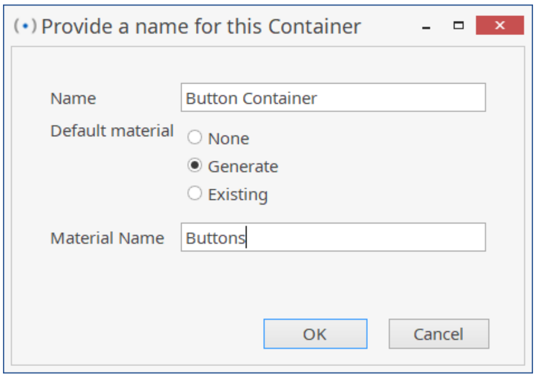


6. Provide a name for the detection and click **OK**.



Note 1

*When adding a container, you can choose leave it empty or to associate a new or existing material to it. You can still change this afterwards in the **Materials** window. For more information, refer to section **4.7 Managing materials** on page 82.*



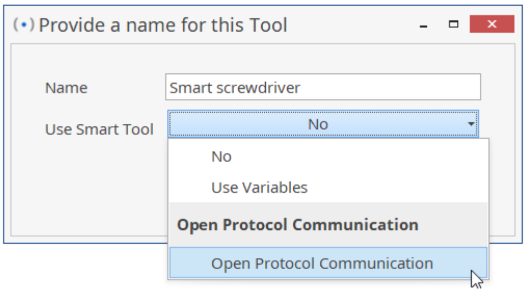
Note 2

*When adding a tool, you should specify whether it is a smart tool or not. In case of a smart tool, you can choose to either use variables of an **Open Protocol** communication module. If you want to use **Open Protocol**, you should add the module first. For more information, refer to section **6.3.3 Adding a communication module** on page 178.*

*Only <u>**one**</u> tool can be linked to a single **Open Protocol** module. To use multiple tools, create multiple **Open Protocol** modules.*

*For more information on configuring your smart tool, refer to section **4.8 Working with smart tools** on page 83.*

*It is always possible to change the option you want to use afterwards. To do so, select the tool in the **Detections** tab and update its **Tool Connection Settings** in the **Properties** tab.*



When using a **Guide** license (=without sensor) a **draw shape** visualization is automatically generated when adding the detection, which will be projected on the workbench when something needs to be done with the object

7. (**Validate** and **Vision** license only) Update the properties in the **Properties** tab if required:
   - **Show error visualizations:**
     If checked, an error projection is shown when the detection is activated at a wrong time in the assembly process, or when an object is not at its expected location. For tools and containers, this feature is enabled by default, for other detections types it is off by default.
   - **External Trigger Variable:**
     It is possible to use an external system, for example a PLC or another sensor than the one provided by Arkite, to validate the presence of a detection. If you want to do so, select the variable containing the information about the detection's presence and the define:
     External Trigger ON State:
     the value that is expected in the External Trigger Variable to activate the detection
     External Trigger ON Time:

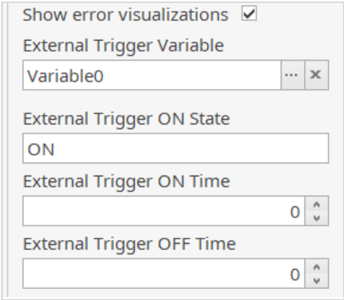
the time in milliseconds that the ON state needs to be reached before the detection is activated.

External Trigger OFF time:

the time in milliseconds the External Trigger variable needs to have another value than the ON state before the detection is deactivated.

Note

In this case, it is not needed to teach the detection. If you do, the teaching will be ignored by the system as long as an external trigger has been configured.



- **Location/Size based on variable:**

It is possible to make the location and size of a detection dynamic, based on the value of variables. For example, you can teach and finetune the detection in the Arkite software but let the location be defined by the input of an external system through the Arkite API.

Note

*For more information on working with the Arkite API, refer to chapter **17 The Arkite API** on 250.*

In the **Location Variables** properties, you can define the variables that contain the X, Y and Z coordinates of the detection (in mm from the origin). These values define the coordinates of the lower left point of the detection's minimum bounding box.

In the **Size Variables** properties, you can define the variables that contain the length, width and height of the detection, i.e. the size in the X, Y and Z direction respectively.

When changing location and size variables properties, all projections linked to the detection as well as the picking boxes for a container will move along.

The **Projection Z** variable allows you to define a different Z value for your projection than the bottom of your box. If multiple projections are linked to the detection, the **Projection Z** variable defines the Z value of the lowest projection. If left blank, the projection's Z value maintains its existing distance to the bottom of the box.

The table below contains some examples of how the Z an projection Z location variables behave:

Before setting the variables	Variable values	After setting the variables
Z value for the box = 50 Z value for the projection = 100	Z = 70 Projection Z = 200	Z value for the box = 70 Z value for the projection = 200
Z value for the box = 50 Z value for projection 1 = 100 Z value for projection 2 = 170	Z = 70 Projection Z = 200	Z value for the box = 70 Z value for projection 1 = 200 Z value for projection 2 = 270
Z value for the box = 50 Z value for projection 1 = 100 Z value for projection 2 = 170	Z = 70 Projection Z = (blank)	Z value for the box = 70 Z value for projection 1 = 120 Z value for projection 2 = 190

Note

*When clicking the **Set Location Variables to Default** button, the values of the variables will be set to the box's coordinates and size as defined in the Arkite software.*

*When clicking the **Set Location Variables to Default** button, the values of the variables will be set to the box's coordinates and size as defined in the Arkite software.*

- **Dmx**

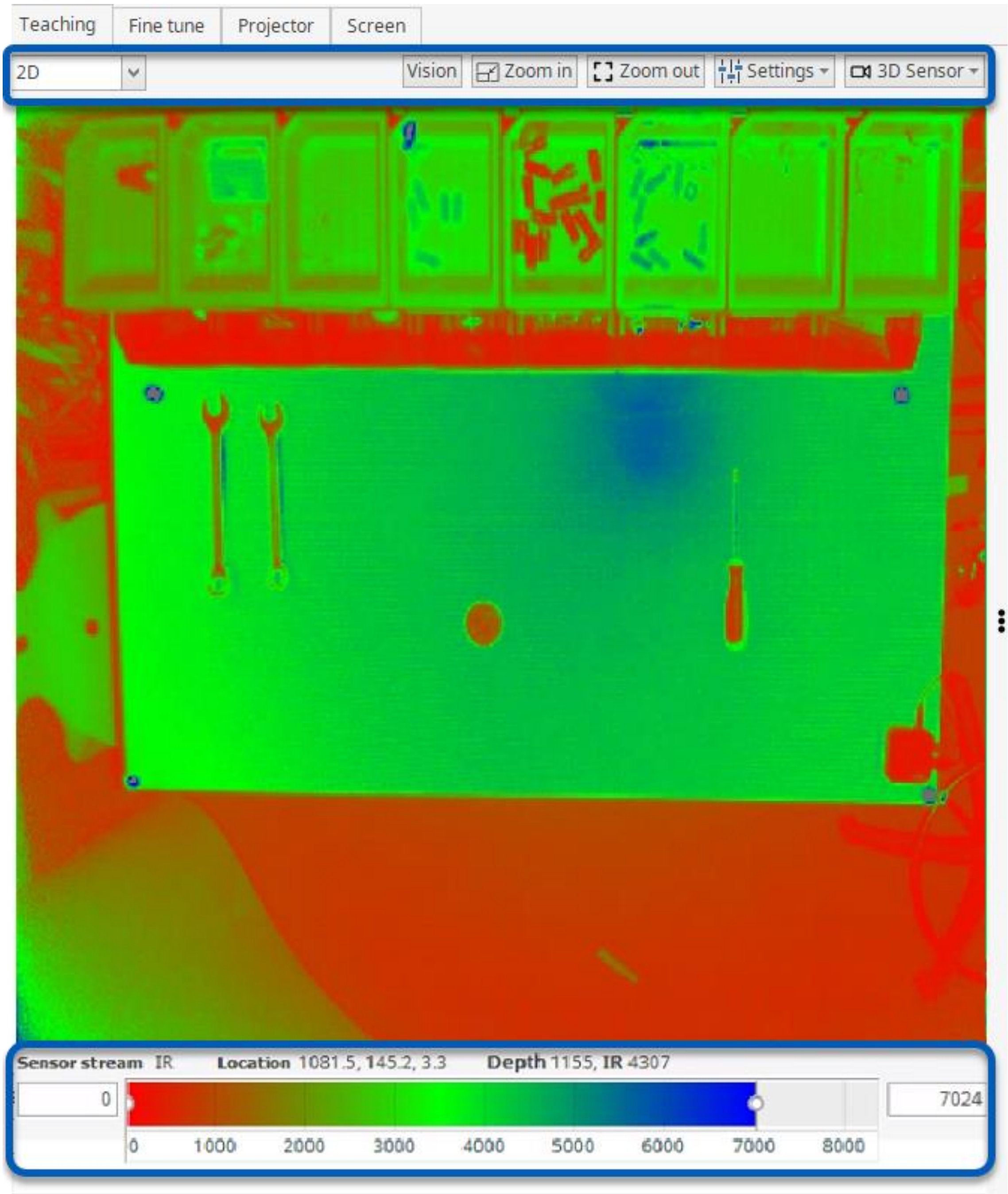
  You can link your detection to a DMX light. To do so, you first need to add a DMX communication module and configure the lights on the connected DMX device. For more information, refer to sections **6.3.3 Adding a communication module** on page 178 and **6.3.4 Configuring lights for a DMX communication module** on page 179.

  In the **DMX Light** dropdown list, you can select the DMX light you want to link the detection to. In the **DMX Color** menu, you can set the color for the linked DMX light. By default, the color of a DMX light that is linked to a detection changes depending on the status of the detection, based on the **Detection colors** template. To manage the behavior and the colors that are used, refer to section **14.2 The Detection Colors template** on page 232.

4.5 Teaching and finetuning detections (Validate and Vision license only)

4.5.1 Optimize your sensor stream view

Before you start teaching detections, you should make sure you have the best view possible on your workbench. To do so, navigate to the **Detections** tab.



In the **Teaching** subtab, optimize your sensor stream view:

- In the dropdown, select the view that gives the best result for your specific situation:
    - **3D:** provides a 3D view of your workbench.

Hold your left mouse button and move around to rotate your view
Use your mouse wheel to zoom in or out
Hold your right mouse button and move around to shift your view to the left or the right

- **2D**: provides a 2D view of your workbench's surface
  Use your mouse wheel to zoom in or out
  Hold your right mouse button and move around to shift your view to the left or the right
- **Top down**: provides an orthographic projection along the Z-axis, so you have a top-down view on your calibration surface without a perspective effect. This view is recommended when you prefer a 2D view, but are working with detections that are high above the calibration surface.
  Use your mouse wheel to zoom in or out
  Hold your right mouse button and move around to shift your view to the left or the right

- **Vision** (Vision license only): shows the Arkite Vision sensor stream, with all boxes that have been taught with the Vision sensor displayed.
- Define the zoom level using the **Zoom in** and **Zoom out** buttons.
- In the **Settings** menu, select the stream of your choice (depth or IR) and mirror the screen if required.
- In the **Sensor** menu, select the sensor that should be used for teaching the detection. The selected sensor is shown on the button:
  - For **Quality Checks**, the Arkite Vision sensor will be selected by default (Vision license only)
  - For all other detections, the Arkite 3D sensor will be selected by default. If you are using a Vision license, you can switch to the Arkite Vision sensor. This can be beneficial when teaching small objects.

    Note

    *By default, the **Infrared - Pseudo color stream** will be shown when the Arkite 3D sensor is selected and the **Infrared – Grayscale** stream will be shown when the Arkite Vision sensor is selected.*
- Use the sliders below the image to optimize the visibility of certain objects. Both the red and blue pegs correspond to a certain infrared value and a green gradient connects both.

  Note

  *Apart from the selection you make in the **Sensor** menu, the settings and options described above are only meant to increase the visibility of the objects for the user. Changing these settings will not have any impact on the detection logic.*

4.5.2 Teaching detections

In the section below, instructions are provided on how to teach different types of detections. You will learn how to:

- Teach an object, container, tool or virtual button
- Teach an activity
- Teach a quality check

Teach an object, container, tool or virtual button

To teach a detection, execute the steps below:

1. Navigate to the **Detections** tab.
2. Select the detection you want to teach to the system.

   Note

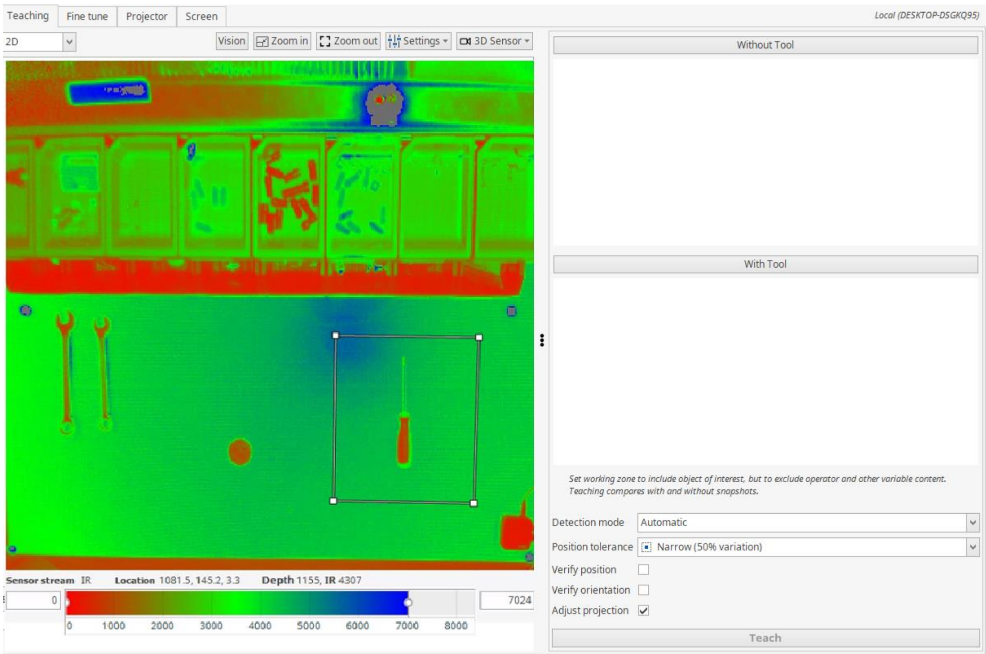
   By default, the Arkite software will select the Arkite 3D sensor for teaching these detections. When you are using a ***Vision*** *license, you can also select the Arkite Vision sensor from the* ***Sensors*** *menu This can be beneficial when teaching small objects.*

   By default, the ***Infrared - Pseudo color stream*** *will be shown when the Arkite 3D sensor is selected and the* ***Infrared – Grayscale*** *stream will be shown when the Arkite Vision sensor is selected.*

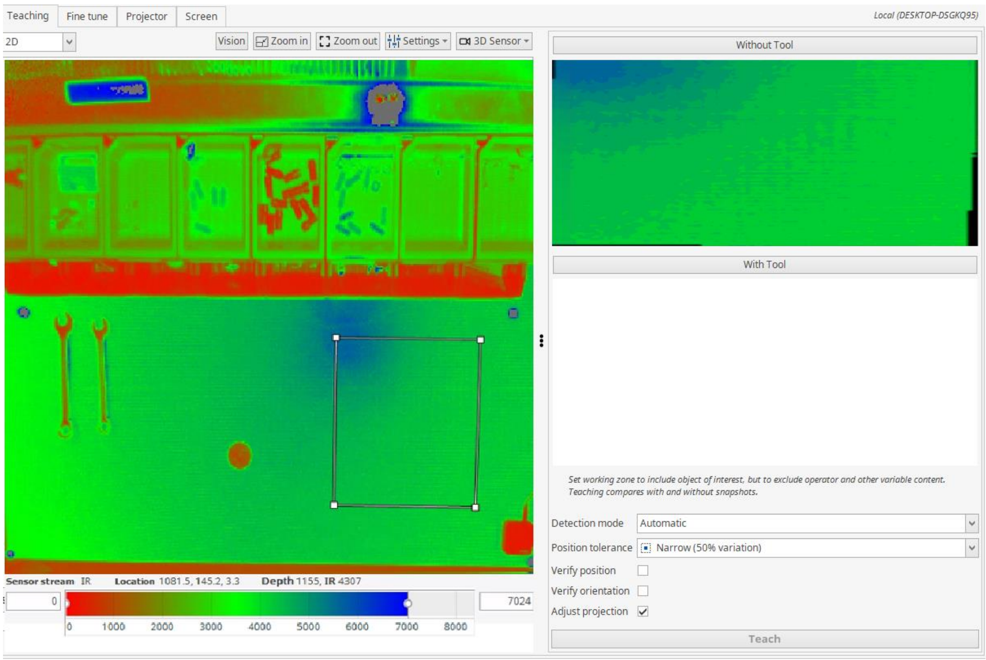
3. Define the area of interest by adjusting the white rectangle. Make sure it does not contain any other variable information than the detection you want to teach.

   Note

   When teaching a tool and using a ***Vision*** *license, you have the option to define a working zone for the location of a tool. The system will then 'follow' the position of that tool within the zone. In such a case, set the white rectangle to the working zone in which the tool can be located.*

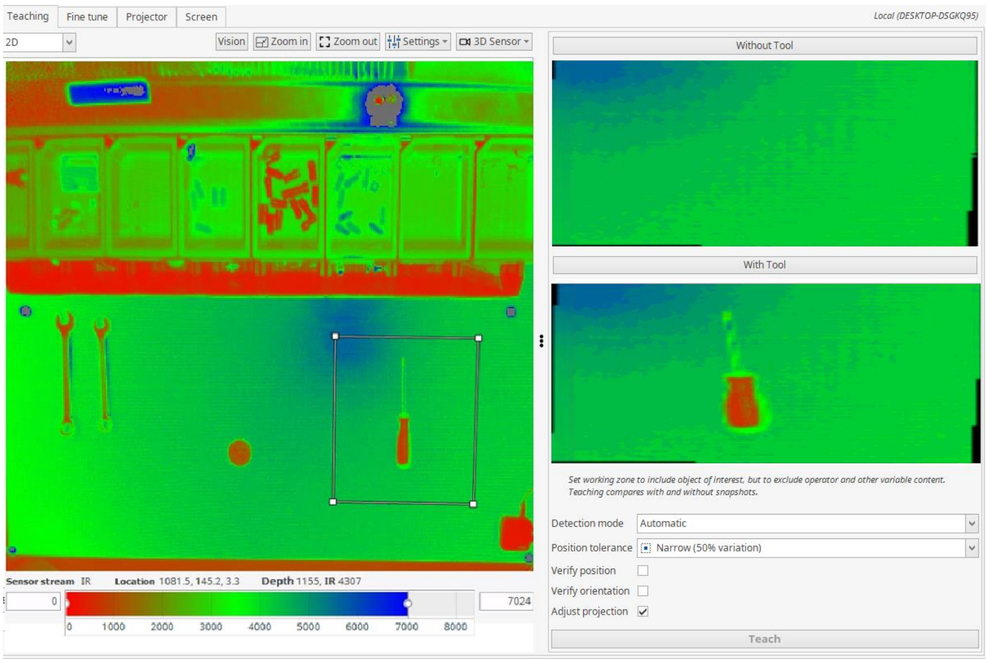


4. Take the detection away and click the **Without...** button.



A snapshot is generated of the area interest, without the detection.

5. Put the detection at its location and click the **With...** button



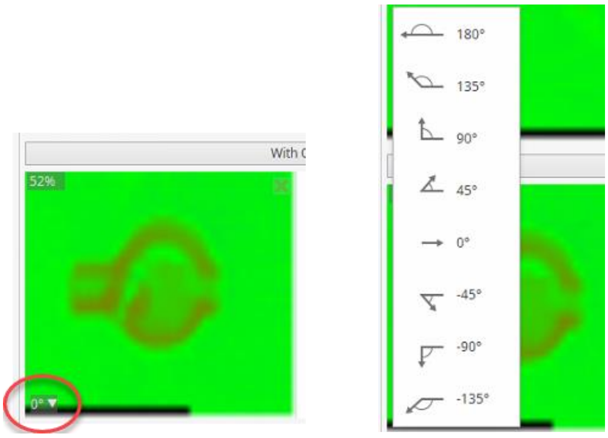
A snapshot is generated of the area interest, containing the detection.

Note

When teaching a virtual a button, use your fist.

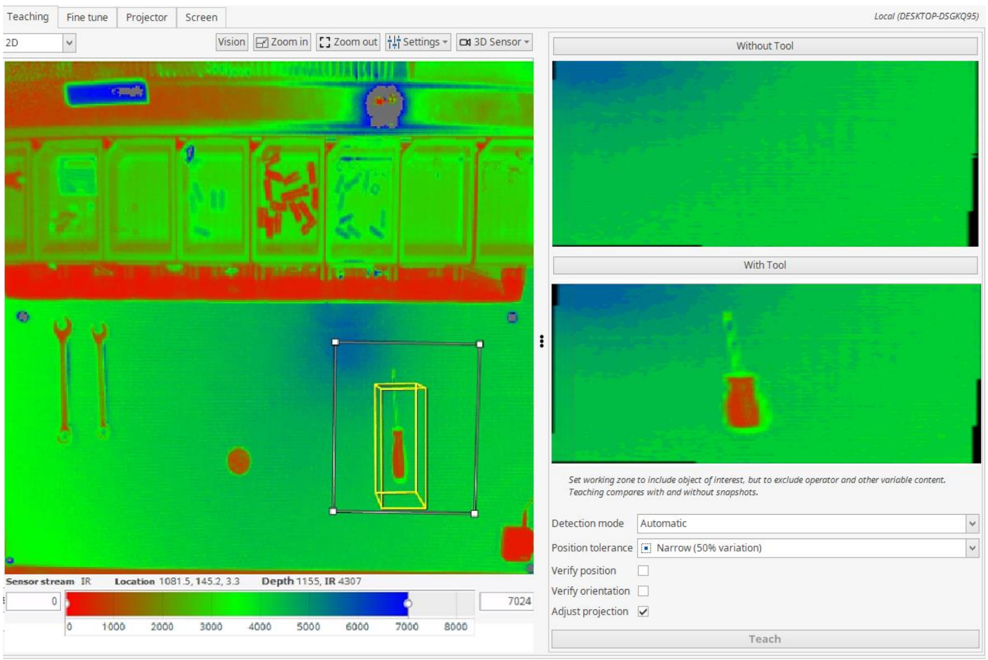
6. Optionally, update the settings:

Setting	Description
Detection mode	Select the sensor that is used for teaching the detection: – **Automatic**: the system will select the stream with the best contrast. This is the default option. – **Depth**: use the depth stream. – **IR:** use the Infrared intensity stream. – **Reflective tape (IR):** in some cases, objects are difficult to detect due to a low contrast or changing light conditions in the working area. This can be solved by replacing the object with a piece of reflective tape with a high and constant IR value. For more information, refer to section **4.9 Working with reflective tape (Validate and Vision license only)** on page 89. – **Infrared edges:** when selecting this option, the IR stream will be used but the sensor will look for edges, in other words for the boundaries of an object where there is a high contrast with the adjacent pixels. It can be used to detect objects regardless their color or variations in infrared lighting conditions. – **RGB:** if the depth and IR stream do not provide sufficient contrast to teach the detection, it's also possible to use the RGB image. If you would like to try and teach detections using this stream, please contact Arkite for support. – **Shape based (Vision license only):** use the shape-based matching algorithm. – **Texture based (Vision license only):** use the correlation-based matching algorithm, based on a Normalized Cross-Correlation (NCC) model. Only available for Quality Check detections. – **Histogram based (Vision license only):** use the histogram-intersection based matching algorithm. Use this option when you have detections that don't have a rigid shape, for example, if you want to detect the presence of a cable that can be bent in various ways, of if you want to detect a dot of glue of which the shape and size may vary. ***Note*** *The histogram based detection mode is not included in the **Automatic** mode. You need to select it explicitly if you want to use it.*
Position tolerance	Define the accuracy of the size of the box that is created after detection. The options are: – **Wide**: the shape that is 100% larger, i.e. twice the size, than the detected object. – **Narrow:** a shape is created that is 50% larger than the detected object. This is the default option when detecting a tool. This way, the operator does not have to put back a tool at the exact same location. – **Accurate**: a shape is created that is 20% larger than the detected object – **Exact:** a shape is created with the same size as the detected object. This is the default option for other detections. – **Search in Zone** (only for Tools and only with **a Vision** license): a shape is created with the size of the working zone. The system will look for the tool anywhere in the zone and keep track of its exact location. This way, the operator can freely decide where to put back a tool after using it, as long as it stays in the zone defined during teaching.
Verify position (Vision license only)	For tools and objects only, and when **Search in Zone** is not selected in the **Position tolerance** property. Check this box if you want the software to check if the object or tool is in the exact same position as during teaching. If it's not, arrows will be projected instructing the operator in which sense he should move the tool or object to reach the correct position.

Setting	Description
Verify orientation (Vision license only)	For tools and objects only. Check this box if you want the software to check if the object or tool is in the same orientation as during teaching. If it's not, a projection will be displayed instructing the operator how to change the orientation. ***Note*** *For objects and tools with an irregular form for which the orientation can vary a lot, it is recommended to add multiple **With Object / With Tool** snapshots from different angles. For each snapshot, indicate the angle by clicking in the lower left corner of the snapshot and selecting the correct angle.* ***Watch out:*** *Rotate the object **counterclockwise** to indicate an increasing positive angle. For example, a quarter-turn counterclockwise results in an angle of 90 degrees, a quarter-turn clockwise results in an angle of -90*.* 
Adjust projection	Define whether projections should be adjusted when teaching the same object a second time.

7. Click **Teach**.

For every detection, a box is created with the same size and position as the detected object. This allows the system to verify whether the object is present.



Important

The system can only detect whether an object is present on the location of the detection. It is not able to 'follow' an object and check whether it is present on another location on the workbench. Therefore, it makes sense to integrate a fixture onto the workbench, where the component can be fixated during the process which makes sure that the entire configuration of components and detections are standardized and stay in the same location over time.

*Only for tools and when using a **Vision** license, you have the option to define a working zone for the location of a tool. The system will then 'follow' the position of that tool within the zone.*

Additionally, a **draw shape** visualization is generated, which will be projected on the workbench when something needs to be done with the object. In case of an error (i.e. the object is not there at a time it should be), the draw shape visualization is projected in red.

For containers, a second box is generated right above the container. This allows the system to detect when something is picked from the container. That works fine, unless you are working with several (rows of) containers stacked upon each other, hiding the boxes from the containers below them to the sensor. For more information on how to deal with stacked containers, refer to section **4.10 Working with stacked containers** on page 89.

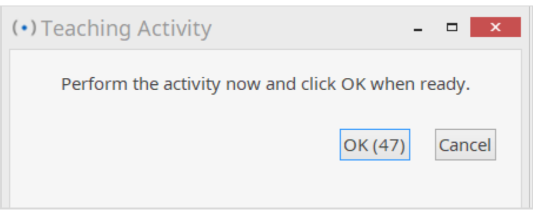
Teach an activity

To teach an activity, execute the steps below:

1. Navigate to the **Detections** tab.
2. Select the activity detection you want to teach to the system.
3. Define the area of interest by adjusting the white rectangle. Make sure it does not contain any other variable information than the detection you want to teach.
4. Optionally, update the settings:

Setting	Description
Detection mode	Select the sensor that is used for teaching the detection: – **Depth**: use the depth stream. – **Reflective tape (IR):** in some cases, objects are difficult to detect due to a low contrast or changing light conditions in the working area. This can be solved by replacing the object with a piece of reflective tape with a high and constant IR value. For more information, refer to section **4.9 Working with reflective tape (Validate and Vision license only)** on page 89.
Adjust projection	Define whether projections should be adjusted when teaching the same object a second time.

5. Click **Teach**.
   The **Teaching Activity** pop up window appears.
6. Perform the activity and click **OK** when done.



A box is created with the same size and position as the area where the activity takes place. This allows the system to verify whether the activity is being executed.

Important

*The system detects movement in the area where the activity takes place. It is not able to verify whether the activity has been executed correctly unless it is performed with a smart tool. In that case, it can use information such as the torque level as an input. For more information, refer to section **4.8 Working with smart tools** on page 83.*

Teach a quality check (Vision license only)

To teach a quality check, execute the steps below:

1. Navigate to the **Detections** tab.
2. Select the quality detection you want to teach to the system.

   Note

   *By default, the Arkite software will select the Arkite Vision sensor for teaching quality checks. Optionally, you can also select the Arkite 3D sensor from the **Sensors** menu. However, most of the time the Arkite Vision sensor will give the best results.*

   *By default, the **Infrared - Pseudo color stream** will be shown when the Arkite 3D sensor is selected and the **Infrared – Grayscale** stream will be shown when the Arkite Vision sensor is selected.*

3. Define the area of interest by adjusting the white rectangle. Set the working zone around the item to check, such as a screw or a clip.

   Note

   When checking multiple objects, create and teach different quality checks for each of them.

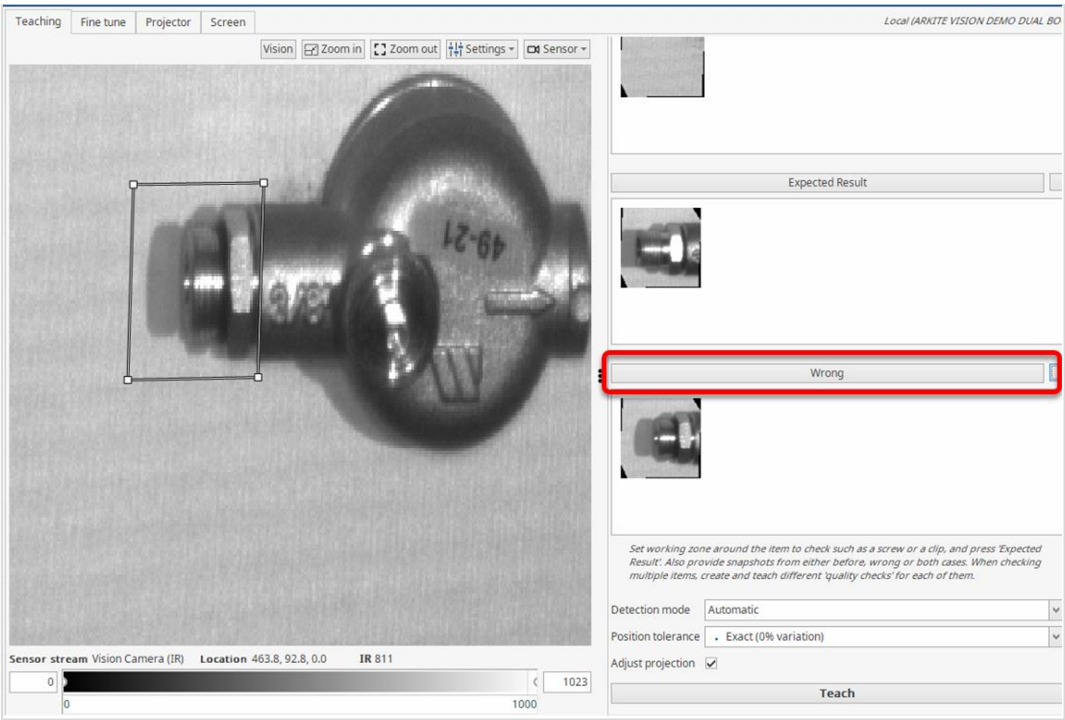
4. Make sure the item to check is not present (i.e. before assembling it) and click the **Before** button.

5. Make sure the item is present and assembled correctly and click the **Expected Result** button. In the example below, a double nipple is entered in the oil filter.

Note

*You can add several **Expected Result** frames if needed.*

6. Optionally, assemble the item in a way that is considered wrong and click the **Wrong** button. In the example below, a double nipple is entered in the oil filter, but in the wrong direction.



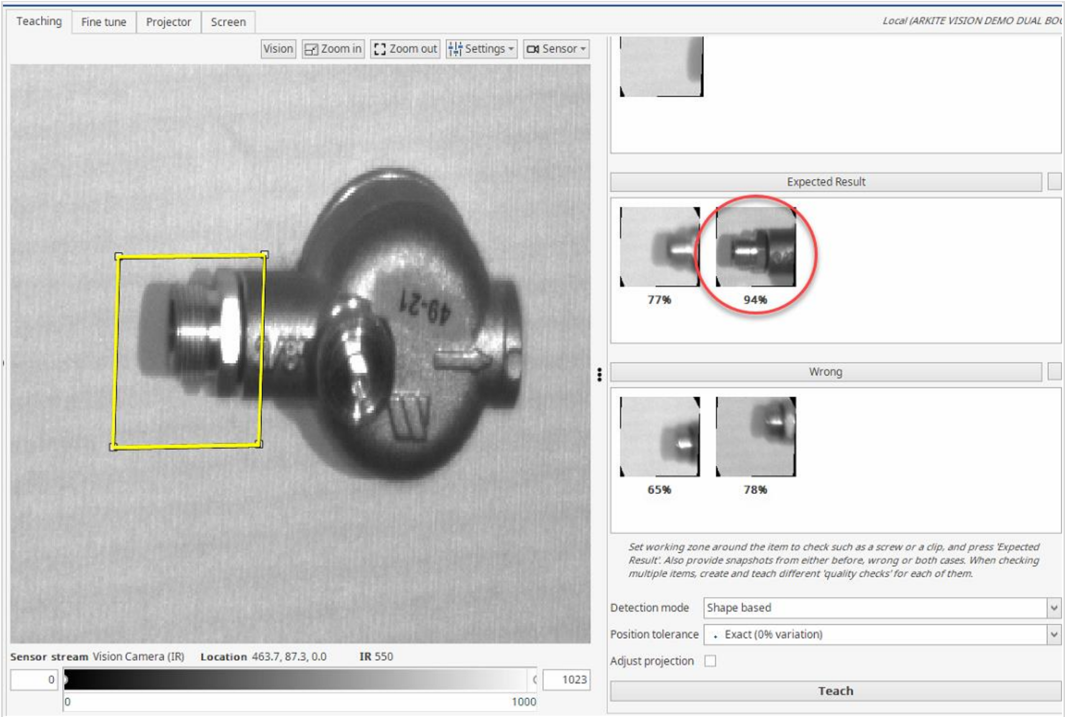
Note

*You can add several **Wrong** frames if needed.*

7. Click **Teach**.

A box is created with the same size and position as the area where quality check needs to be performed. When performing the quality check, the system will compare the live image with all **Before**, **Expected Result** and **Wrong** frames. The quality check will be considered successful when the live image's similarity is the highest for (one of the) **Expected Result** frame(s) and more than the minimum percentage as defined in the **Detection Rate** property on the **Finetune** tab.

In the example below, a quality check will be successful because there is a 94% similarity with one of the **Expected Result** frames, and there is no higher similarity with one of the other frames.



Additional to the detection box, an occluder box will also be generated using the Arkite 3D sensor. If the occlusion rate exceeds the minimum rate, the system will wait to evaluate the quality check until the occlusion is gone. This prevents a quality check from being classified as wrong while an operator is still working on it. In case the result is not satisfactory, the occluder box can be edited on the **Finetune** tab.

Teach detections based on a recording

In order to save time or to work away from the workbench, you can make a recording of all detections and add them later based on the recording. For more information about recordings, refer to chapter **9 Recordings** on page 202.

4.5.3 Test a detection

After a detection has been taught, always test whether the teaching has been successful:

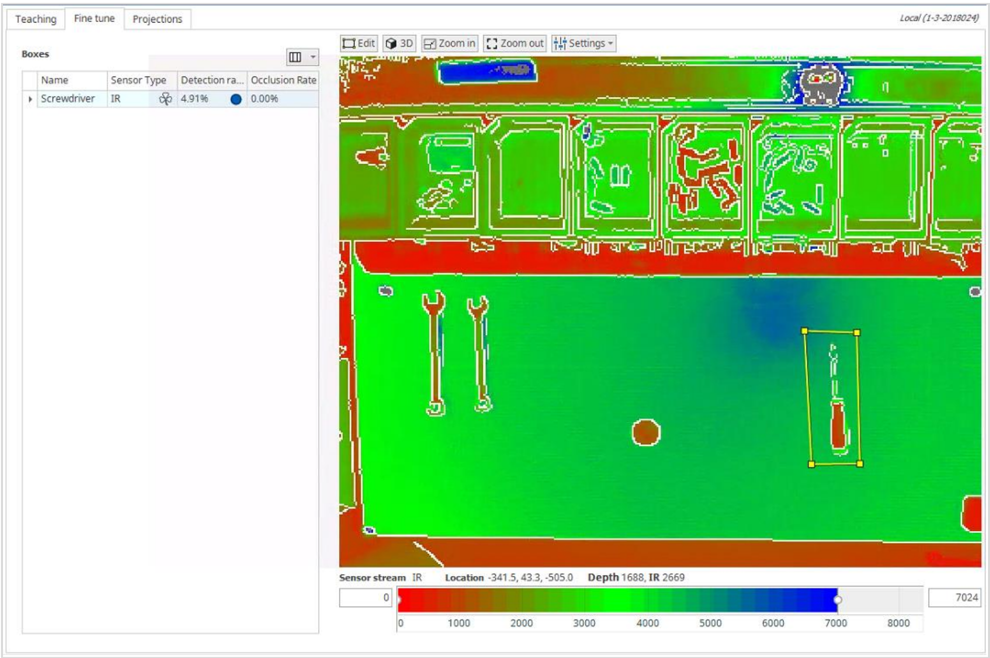
- Place the object or tool and take it away and check if the state of the detection gets updated.
- Pick material from a container and check if the system detects it.
- Press a virtual button and check whether the system detects this.

If you are not satisfied, you can either reteach the detection or finetune the detection.

4.5.4 Finetune a detection

After a detection has been taught, you can tweak the boxes that have been generated in the **Finetune** subtab.

1. Under **Detections**, select the detection you want to finetune.
2. In the **Finetune** subtab, select the box you want to finetune.
3. Click the **Edit** button above the image.
4. Change the size and position of the box by dragging the angular points with your mouse. The same can be achieved by updating the coordinates in the **Properties** panel.

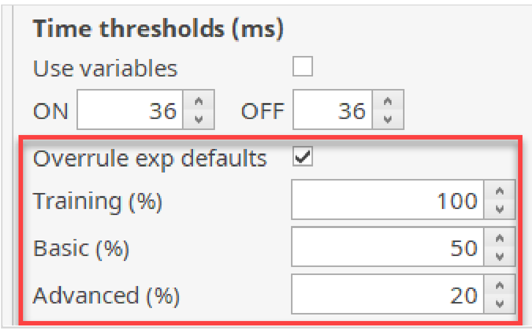


Tip

*Right-click an angular point and select **Remove Point** to delete it or **Insert Point** to add an additional angular point.*

5. Update the properties of the box in the **Properties** panel:

Property	Description
Z-Coordinates	Change the position of the box in relation to the workbench by updating the **Base** z-coordinate or the **Height** of the box.
Calculation properties for detections taught with detection mode Depth or IR	
Use IR	Check the checkbox to use the **Infrared** stream. Uncheck the checkbox to use the **Depth** stream.
Image processing	Select an image processing filter if needed: – **Background subtraction (depth stream)**: filters away all background, so only the moving foreground objects are visible. – **Edge detection (IR stream)**: the sensor will look for edges, in other words for the boundaries of an object where there is a high contrast with the adjacent pixels. It can be used to detect objects regardless their color or variations in infrared lighting conditions.

Property	Description
IR intensity	Define the minimum and maximum IR intensity in the box that is required to consider an object as present.
Detection rate	Define the minimum and maximum percentage of fill rate (or edge rate when Edge detection is used) of the box that is required to consider an object as present.
Calculation properties for detections taught with detection mode Shape Based, Texture Based or Histogram Based (Vision license only)	
Calculation Type	Algorithm that is used for calculating the detection: Shape Based, Texture Based or Histogram Based
Search in zone	Check if you want the system to look for the tool anywhere in the zone and keep track of its exact location.
Detection rate	Define the minimum and maximum percentage of fill rate of the box that is required to consider an object as present. For Quality checks, this is the minimum percentage of similarity that is needed to classify the detection as correct.
Position Tolerance (mm)	Define the difference in position that is acceptable. Only available when **Verify Position** has been selected when teaching the detection.
Orientation Tolerance (°)	Define the difference in orientation that is acceptable. Only available when **Verify Orientation** has been selected when teaching the detection.
Occlusion	Check the **Occlusion logic** checkbox if you want the system to verify if the object is hidden. The **Min Rate** dropdown list contains the minimal percentage of occlusion that is required to consider the object as hidden.
Time thresholds (ms)	Define for how long an object needs to be within (ON) or outside (OFF) the range defined in the fill rate properties to change the detection state. For activities, you can disable the OFF setting. To do so, uncheck the **Wait for OFF** checkbox. In this case, an activity does not need to be deactivated to proceed to the next step when used in a process. In case the **Experience Level** functionality is enabled, a percentage of these time thresholds is used for every specific experience level. ***Note*** *For more information on the **Experience Level** functionality, refer to chapter **7 Experience levels** on page 194.* To overrule the default percentages for this specific detection, check the **Overrule exp defaults** checkbox. After checking this checkbox, the system will fall back to 100% for level 1, 50% for level 2 and 20% for level 3. You can update these percentages as required.  To change the default percentages for all detections, refer to section **7.3 Experience level settings** on page 195.

Property	Description
Coordinates	Change the position of the box by updating the X and Y coordinates of the angular points. ***Note*** *You can also change the size and position of the box by dragging the angular points with your mouse.*

4.5.5 Managing visualizations for the detection

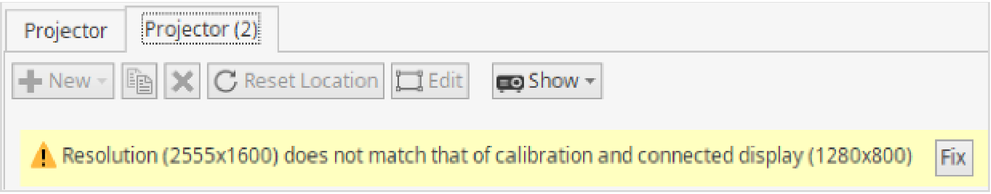
What?

After a detection has been added, you can add visualizations to it on any of the available display subtabs. You will see a subtab for every display that is available:

- If you are starting from a blank project, only the **Projector** subtab will be available.

  Note

  *When you have created the project before calibrating, a **Projector** display is added with the screen resolution of the device you are working on (e.g. your laptop). This may not match with the actual projector resolution, used during calibration. This will cause projections to be faulty. A message on the **Projector** subtab will warn you for this. Click **Fix** and update the **Resolution** property to the correct value in the projector's **Properties** pane.*

- If you are starting from the default template, a **Projector** and a **Screen** subtab will be available.

  Note

  *To add more displays, refer to section **6.2.3 Adding a display** on page 161.*

The system will automatically generate a draw shape visualization on the **Projector** subtab:

- When using a **Validate** or **Vision** license (= with Arkite 3D sensor), this is done when teaching the detection.
- When using a **Guide** license (= without sensor), this is done when adding the detection.

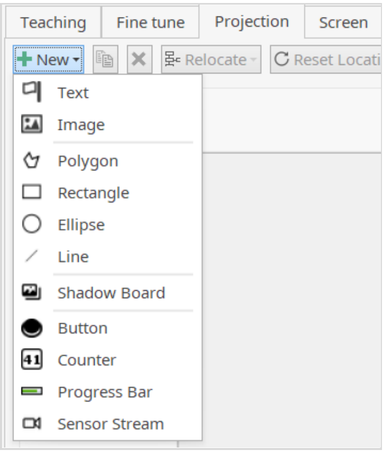
You can edit and update that visualization.

Adding visualizations

To add a new visualization, execute the steps below:

1. Under **Detections**, select the detection for which you want to add a visualization.

2. Click the display subtab on which you want to add a visualization, **Projector** or **Screen**.
3. Click the **New** button and select the visualization of your choice.



Tip

*You can also select an existing visualization and click **Duplicate**.*

4. Define the size and position of the visualization using your mouse.

Tip

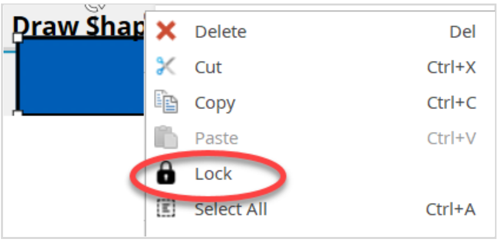
While dragging the visualization with your mouse, visual guidelines will appear that will allow you to easily align the visualization. The guidelines below will appear:

- *Guidelines towards the **middle of the screen** will appear when you reach the middle of the screen with an edge or with the middle of the visualization, both horizontally and vertically.*
- ***Object to object alignment lines** appear to help you to align two visualizations with each other, both horizontally and vertically. This way, you can align the top, the bottom or the middle of the visualizations with each other.*
- *Arrows appear to help you to position two visualizations at the **same distance from the edge or the center** of the screen.*
- *When you have more than two visualizations, arrows appear when the distance between visualizations 1 and 2 is the same as between visualizations 2 and 3. This helps you to ensure there is the same distance between visualizations.*
- *When you are changing the size of an object, lines appear inside the objects when they have the same length or width. This helps you to make the objects have the **same size**.*

5. In the **Properties** pane, update the properties of the visualization if needed.

Note

- *By default, the visibility and the color of a visualization that is linked to a detection and that is shown on a **Projector** display changes depending on the status of the detection, based on the **Detection colors** template. To manage the behavior and the colors that are used, refer to sections **5.5.7 Managing projection settings** on page 136 and **14.2 The Detection Colors template** on page 232.*
- *The properties that are available depend on the type of visualization you have added. For an overview of visualization types and their properties, refer to sections **5.5.1 Visualization types** on page 116 and **5.5.2 Visualization properties** on page 117.*
- *To protect a visualization from being moved or edited by accident, right-click it and select **Lock**.*

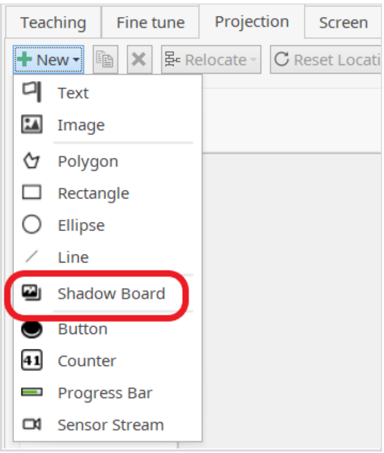


Adding a shadow board for a tool

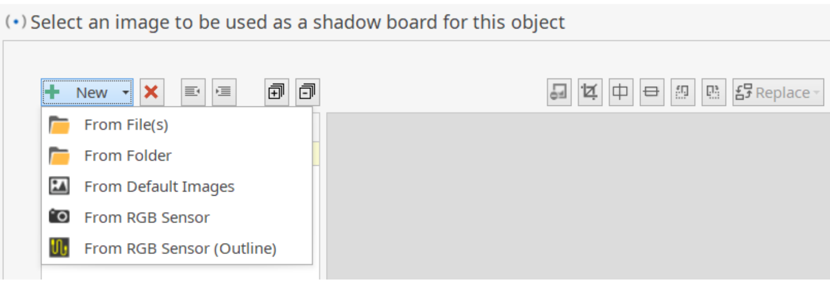
For tools, a shadow board visualization is typically added. This is an image that is projected at the location of the tool when it's not there. This way, the operator always knows where to put the tool back after use.

To add a shadow board, execute the steps below:

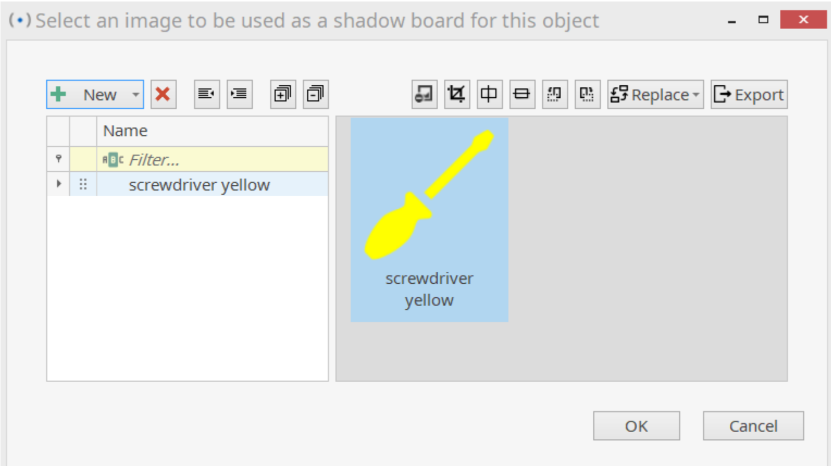
1. Under **Detections**, select the tool for which you want to add a shadow board.
2. In the **Projector** subtab, click **New** and select **Shadow board**.



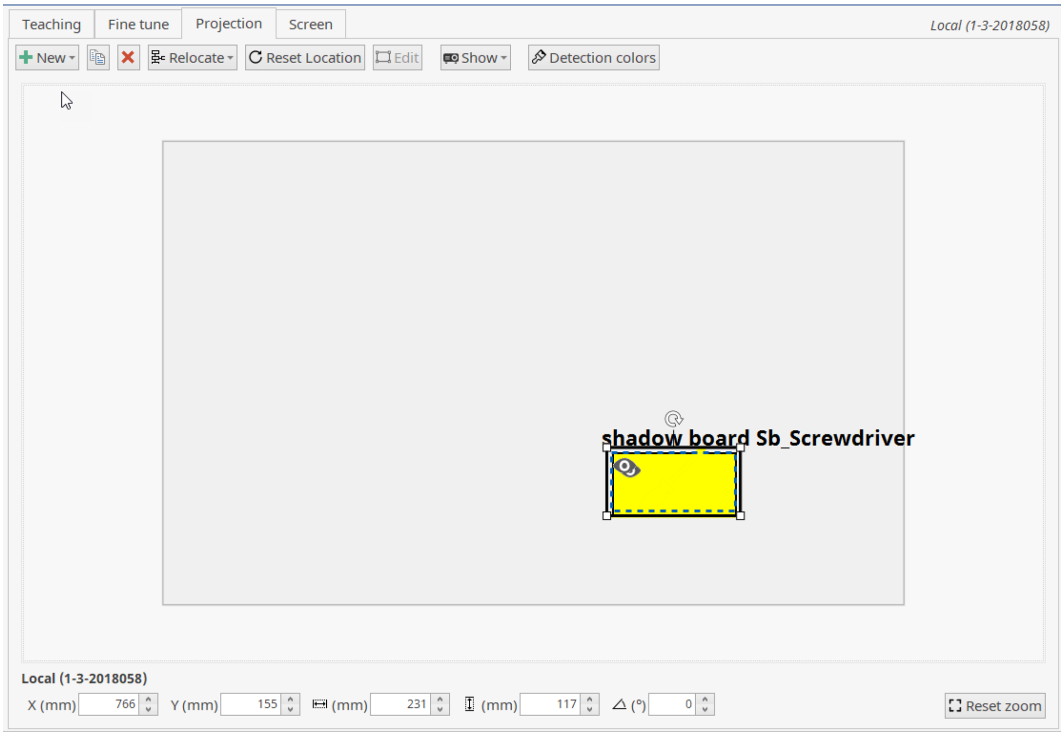
3. Add an image to use for the shadow board. By default, no images will be available in your project. Click the **New** button and choose to add an image from your pc or by using the sensor's RGB camera. Under the option **From Default Images**, you will find a number of images for the most common tools.



4. Select the image and click **OK**.



The shadow board is added at the same location as the default projection for the tool.

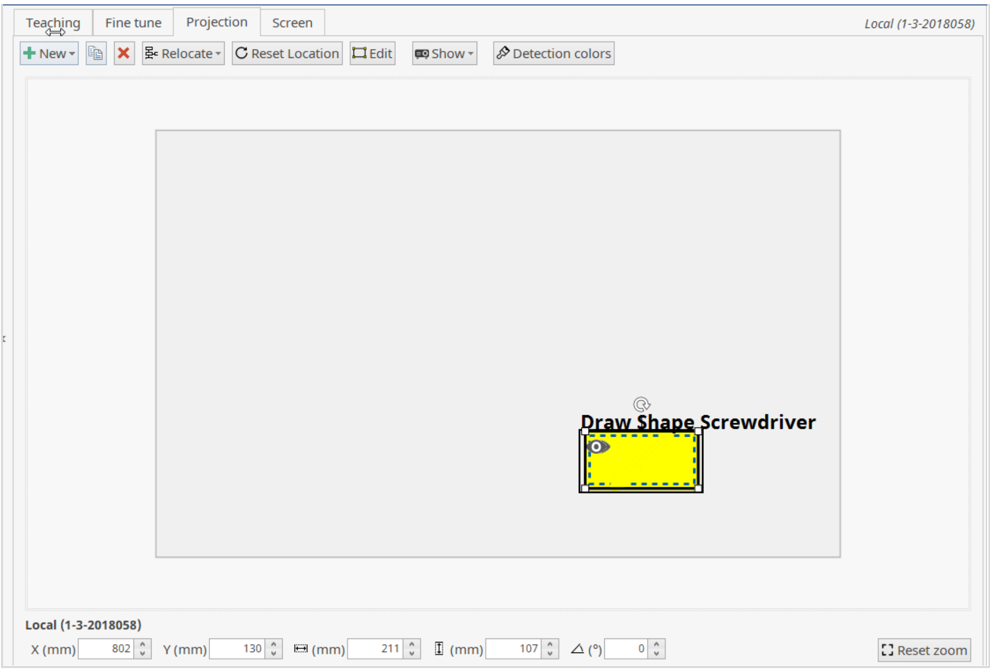


Editing visualizations

To edit an existing visualization, execute the steps below:

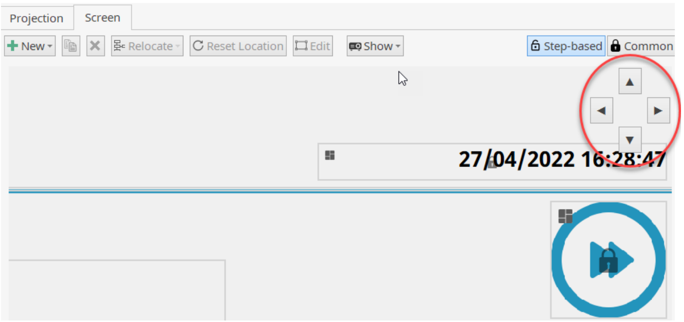
1. Under **Detections**, select the detection for which you want to edit a visualization.
2. Click the display subtab on which you want to edit a visualization, **Projector** or **Screen**.

3. Select the visualization you want to edit.



Note

- *You can consult multiple visualizations by holding the* ***Ctrl*** *button and clicking the visualizations of your choice with your mouse, or by holding the left mouse button and dragging a rectangle over the visualizations of your choice. Press* ***Ctrl-A*** *to select all visualizations.*
- *You can zoom in and out using your mouse wheel. When zoomed in, arrows appear to navigate around in your display.*



4. For draw shapes:
   - Change the size and position of the visualization by dragging the shape or the angular points with your mouse.
   - To change the shape of the visualization, click the **Edit** button above the image and move the angular points.
5. For all shapes, update the properties of the visualization in the **Properties** pane if needed.

Note

The properties that are available depend on the type of visualization you have added. For an overview of visualization types and their properties, refer to sections ***5.5.1 Visualization types*** *on page 116 and* ***5.5.2 Visualization properties*** *on page 117.*

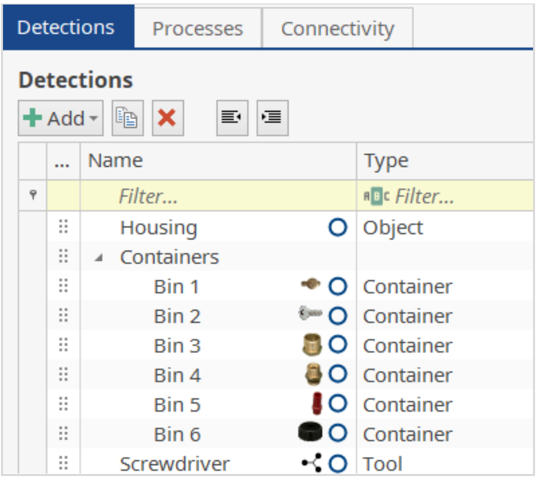
To delete an existing visualization, select it and click **Delete**.

By default, the visibility and the color of a visualization that is linked to a detection and that is shown on a **Projector** display changes depending on the status of the detection, based on the **Detection colors** template. To manage the behavior and the colors that are used, refer to sections **5.5.7 Managing projection settings** on page 136 and **14.2 The Detection Colors template** on page 232.

4.6 Managing detections

What?

After having added the required detections, you can manage and organize them to have a better overview.



How to manage detections

On the **Detections** tab, you can:

- Select a detection and click to duplicate it.
- Select a detection and click to delete it.
- Select a detection and drag it with your mouse to change its order.
- Organize your detections:
  - Click **New** and select **Group**.
    A group is created. Any new detection you add while this group is selected will be added under the group. You can also drag existing detections into the group.
  - Select one or more detections and click the button. This way, a group can also be created.
- Click the button to increase the level of detail you see.
- Click the button to decrease the level of detail you see.

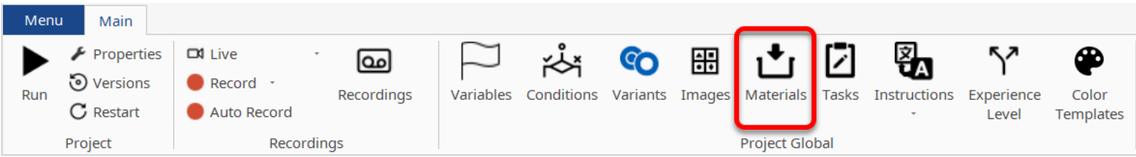
4.7 Managing materials

What?

When creating a container, you can either leave it empty, associate it with a new material or select an existing material. These materials can still be managed afterwards.

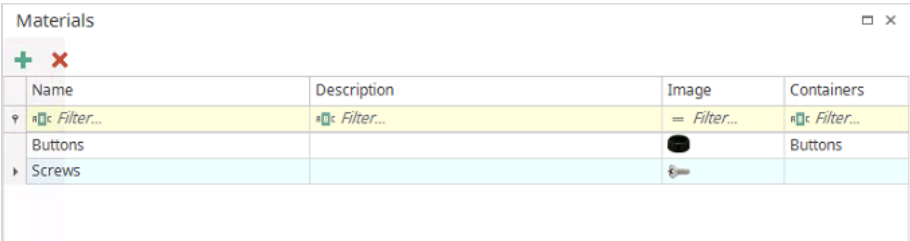
How to manage materials

In the ribbon, in the **Project Global** group, click the **Materials** button.



The **Materials** window appears. For every existing material, you can update the existing information by clicking in the column of your choice:

- **Name:** change the material's name
- **Description:** change the material's description
- **Image**: add an image of the material. This image will be used on the **Detections** tab.
- **Container:** associate the material to one or more containers. You can only select containers that haven't been associated with a material yet.



Click ➕ to add a new material to the list.

Click ❌ to remove a material from the list.

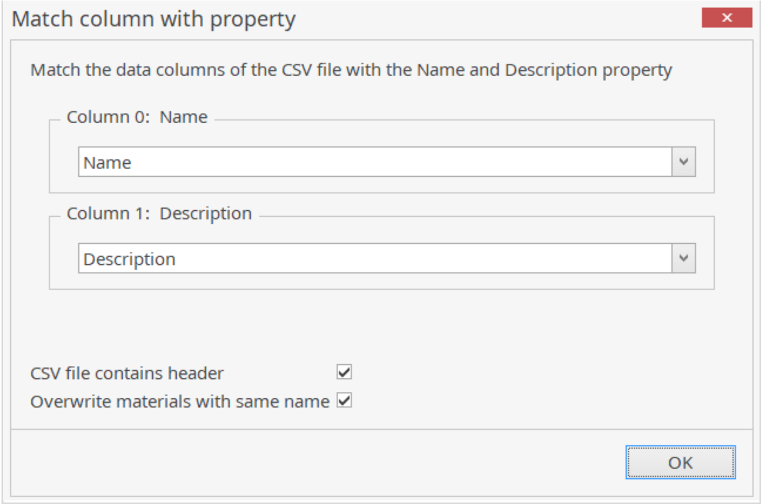
Importing and exporting materials

To export your list of materials to a CSV file, click the button. The **Name** and **Description** column will be exported.

You can also add materials to your list by importing a CSV file. To do so:

1. Click the button.

2. In the **Match column with property** window, you can indicate which column from the CSV file you want to use for the material **Name** and **Description**.



Note

- *If the top row of the CSV file contains a value **Name** and/or **Description**, the Arkite software will suggest to match these columns with the corresponding property.*
- *If you check the **CSV file has header** checkbox, the upper row of your CSV file will not be imported.*
- *If you check the **Overwrite materials with same name** checkbox, existing materials will not be duplicated. The information in the Description column will be overwritten with the information from the CSV file.*
  *If you don't check the **Overwrite materials with same name** checkbox, materials with a name that already exists will be added. A **(2)** is added behind the name of the newly added material.*

3. Click **OK**.

The materials from the CSV file will be imported into your project.

4.8 Working with smart tools

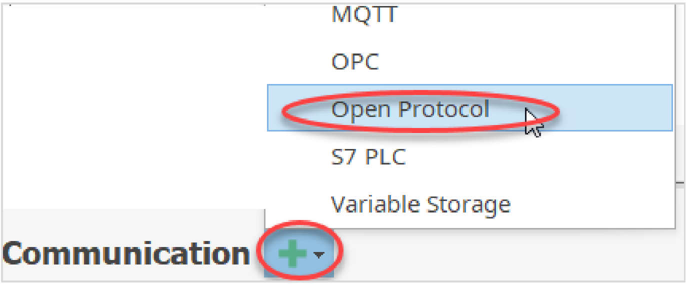
What

Smart tools are tools that can communicate with external systems, such as the Arkite software. Through the Arkite software, they can be configured in such a way that they can only be used during the correct step in the assembly process. When using a **Validate** or **Vision** license (= with Arkite 3D sensor), the system will also ensure the smart tool can only be used at the correct location.

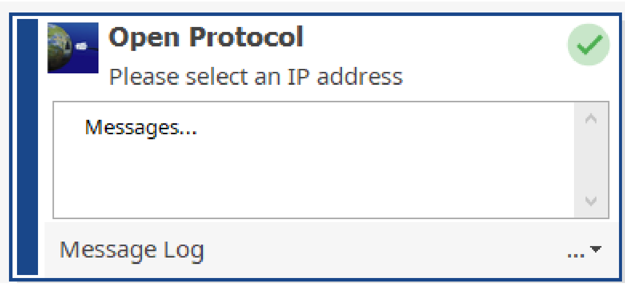
How to configure smart tools using Open Protocol communication

To configure a smart tool using Open Protocol communication, execute the following steps:

1. Make sure your smart tool is on the same network as your workstation.
2. In the **Connectivity** tab under **Communication**, click the **Add** button and select **Open Protocol**.



A new card for the **Open Protocol** communication module is added to the list of communication modules.

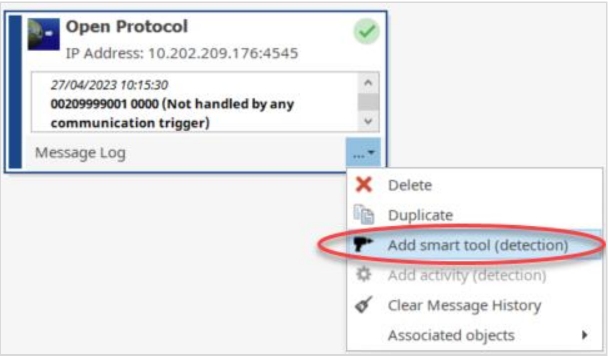


3. In the **Properties** pane, configure the properties for the communication module.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API.*
Comment	Optionally, write a few words of explanation regarding the use of the communication module.
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged.
Automatic Smart Tool Settings	If this checkbox is enabled, the required variables for communication with a smart tool will be generated and configured.
Device Type	– When using Atlas Copco's MicroTorque Controller MTF6000, select **MTFocus 6000**. – For all other devices, select **Generic**.
IP Address	The IP address of the remove device, for example: 10.0.11.12.
Port	The TCP/IP port to use. Default for open protocol communication is 4545.
Start communication revision	When an open protocol communication actuator is first sparked, a 'start communication' message will be sent to the controller. This message contains a 3 digit revision. For most controllers, this revision is '001'. Check the manual of your controller to check for divergent revisions.
End communication revision	When the module is paused, an 'end communication' message will be sent to the controller. This message contains a 3 digit revision. For most controllers, this

Property	Description
	revision is ‘001’. Check the manual of your controller to check for divergent revisions.
Stop String Type	Every message will be ended by a special ‘ending character’. The default value is null (null character). Other types include a CRLF (carriage return line feed) and #.
Log ‘Keep Alive’ messages	Every incoming, outgoing and exception message will be logged. However, to keep the connection with the controller open, ‘keep alive’ messages will be sent (and received) every ten seconds. In most situations, you don’t want to see these messages cluttering your message stream. However, if you would like to see these messages, please check this checkbox.
Open Protocol Device Test	Before an open protocol device can be used, a parameter set must always be selected. Provide a parameter set in the text input field (typically consisting of three digits, e.g. 000, 001, 002....) and click **Set Parameter**. Once you have done that: – Click the **Enable Tool** button to enable the tool. – Click the **Disable Tool** button to disable the tool. By clicking these buttons, the communication towards the tool can be tested.

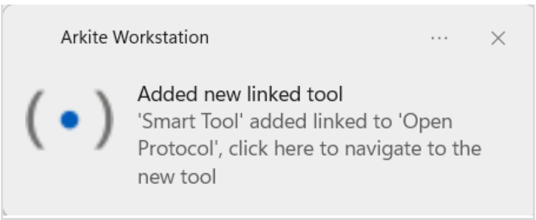
4. Select the card for the **Open Protocol** module, click the **...** button in the lower-right corner of the card and select **Add Smart Tool (detection)**.



Note

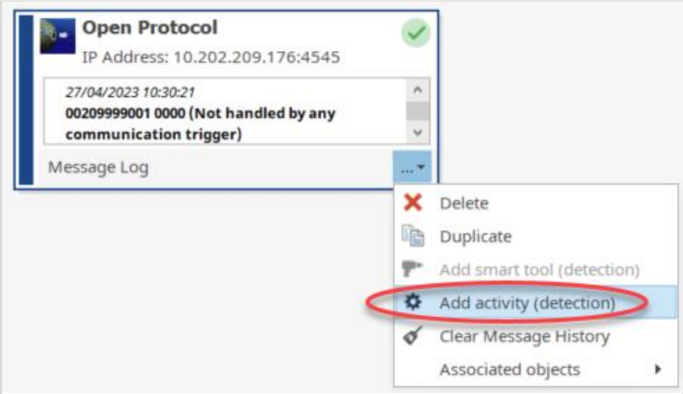
*Only __one__ tool can be linked to a single **Open Protocol** module. To use multiple tools, create multiple **Open Protocol** modules.*

5. A smart tool linked to the **Open Protocol** module is created on the **Detections** tab. Click the toast message in the lower-right corner of the screen to navigate to it.

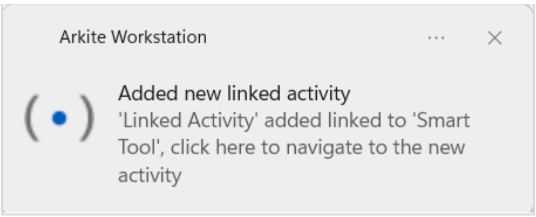


6. (optional) When using a **Validate** or **Vision** license, teach the tool as described in section **4.5 Teaching and finetuning detections (Validate and Vision license only)** on page 64.
This way the system can detect when the tool is taken away or placed back.

7. On the **Connectivity** tab, select the card for the **Open Protocol** module, click the **...** button in the lower-right corner of the card and select **Add Activity (detection)**.



8. An activity linked to the smart tool you set up in step 5 is created on the **Detections** tab. Click the toast message in the lower-right corner of the screen to navigate to it.



9. (optional) When using a **Validate** or **Vision** license, teach the activity as described in section **Teach an activity** on page 70.

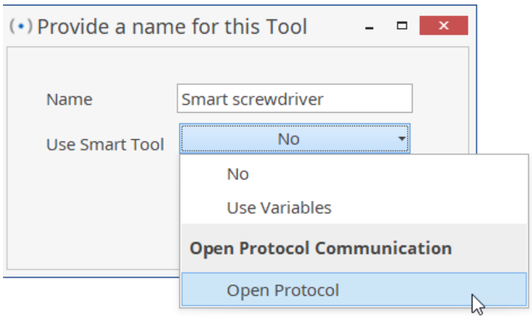
   This way the system can detect when the tool is being used at the desired location.

10. Add more activities as needed.

How to configure smart tools using variables

To configure a smart tool using variables, execute the following steps:

1. Navigate to the **Detections** tab.
2. Click the **New** button and select **Tool.**
3. Specify if you want to use variables or an **Open Protocol** communication module.



Note

If you want to use Open Protocol communication, you should add the module first and leave the ***Automatic Smart Tool Settings*** *checkbox enabled. For more information, refer to section* ***6.3.3 Adding a communication module*** *on page 178.*

*It is always possible to change the option you want to use afterwards. To do so, select the tool in the **Detections** tab and update its **Tool Connection Settings** in the **Properties** tab.*

4. (optional) When using a **Validate** or **Vision** license, teach the tool as described in section **4.5 Teaching and finetuning detections (Validate and Vision license only)** on page 64.

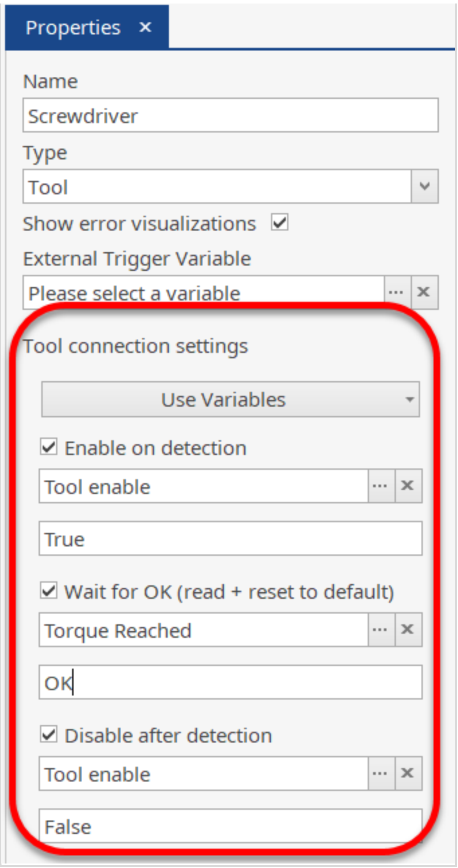
   This way the system can detect when the tool is taken away or placed back.

5. Select the tool in the **Detections** tab and configure the tool connection settings in the **Properties** pane:

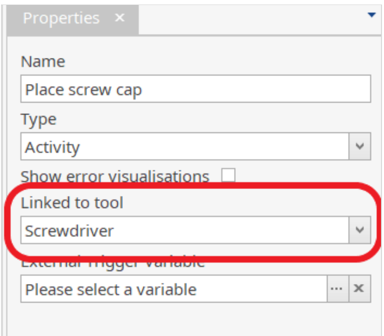
   Note

   *If you are using an **Open Protocol** communication module, the required variables and settings will be preconfigured by the Open Protocol module.*

Property	Explanation
Enable on detection/step	The selected variable will be set to the specified value when: – the associated activity detection becomes active (**Validate** or **Vision** license) – the associated process step is reached (**Guide** license) ***Example*** *Set the variable **tool enable** to **true** when the activity on time or process step is reached.*
Wait for OK (Validate license only)	When this setting is configured, the activity will only be regarded as completed when the selected variable reaches the selected state. Afterwards, the variable will be set to its default value, making sure that one positive trigger of the variable can only confirm one activity. ***Example*** *Wait until the variable **torque reached** is **true** to complete the activity.*
Disable after detection/step	The selected variable will be set to the specified value when: – the associated activity detection becomes inactive (**Validate** or **Vision** license) – the associated process step is completed (**Guide** license) ***Example*** *Set the variable **tool enable** to **false** when the activity off time is reached or when the activity process step is completed.*



6. Link the tool to an activity:
   - Add the activity as described in section **4.4 Adding detections** on page 60.
   - (optional) When using a **Validate** or **Vision** license, teach the activity as described in section **Teach an activity** on page 70.
     This way the system can detect when the tool is being used at the desired location.
   - Select the activity in the **Detections** tab.
   - In the **Properties** pane, link the activity to the tool:



4.9 Working with reflective tape (Validate and Vision license only)

What

In some cases, objects are difficult to detect due to a low contrast or changing light conditions in the working area. This can be solved by replacing the object with a piece of reflective tape with a high and constant IR value.

How to use reflective tape – option 1

1. Replace the object on your working bench with a piece of reflective tape of the same size.
2. Teach the system the location of this reflective tape, using the **Sensor Mode** Reflective tape.
   For more information, refer to section **4.5.2 Teaching detections** on page 65.
3. On the **Finetune** tab, change the **Fill Rate** fields:
   When working with reflective tape, the object is present when the reflective tape cannot be detected. For example, if a 70-100% range is used to detect if the reflective tape is present, you should change the range to 0-70% in order to let the system detect correctly the reflective tape is covered by the present object.
   For more information, refer to section **4.5.4 Finetune a detection** on page 74.

How to use reflective tape – option 2

1. Replace the object on your working bench with a piece of reflective tape of the same size.
2. Teach the system the location of this reflective tape, using the **Sensor Mode** Reflective tape.
   Use the snapshot with the tape in the **Without** image.
   Use the snapshot without the tape in the **With** image.
   For more information, refer to section **4.5.2 Teaching detections** on page 65.

4.10 Working with stacked containers (Validate and Vision license only)

What?

In many cases, several picking containers will be stacked upon each other on the workbench. Except for the top row, the containers will therefore be hidden from the sensor's view and the system won't be able to see them. In this chapter, a workaround for this issue is explained.

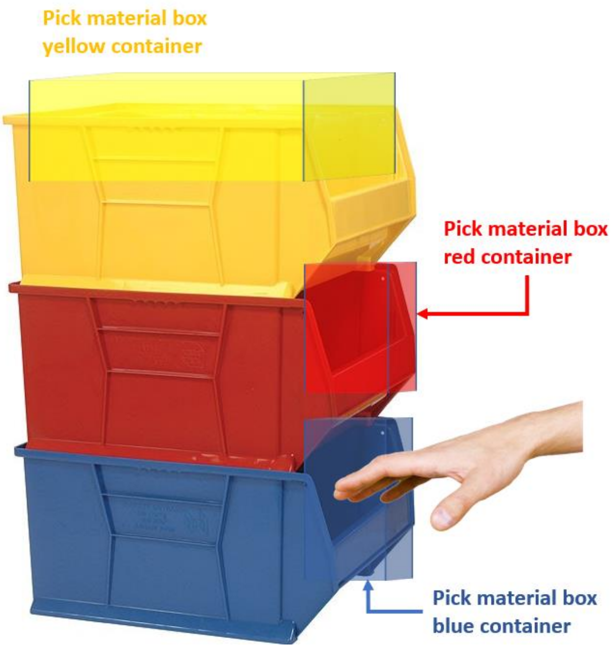
How to work with stacked containers

When a container is taught, the system automatically creates two boxes:

- A box to detect if the container is present
- A box to detect right above this box to detect the operator's hand when picking material from the container.

When working with stacked containers, the boxes for all containers will by default be all on top of each other, and the system will only be able to detect the upper row of containers. This is not really a problem for detecting container presence, since they will only be removed very rarely.

However, to successfully detect picking materials, you should move the corresponding boxes **in front of** the actual containers and make them a lot **smaller**. You should set them in such a way that they form a **small cuboid gate** at the entrance of the container. As a result, the boxes won't be occluded to the sensor and the system can successfully detect the operator's hand going into the correct container and picking a material. For more information on how to edit the pick material boxes, refer to section **4.5.4 Finetune a detection** on page 74.



4.11 Working with mobile picking racks (Validate and Vision license only)

What?

In complex assembly processes, many materials are required to complete an assembly. In such a case, containers are often organized in different mobile picking racks that are exchanged during the assembly so that the relevant materials are available at all times.

It is important the Arkite Operator Guidance system only calculates the presence of the containers in the active picking rack, to avoid error visualizations for containers that are not there.

How to work with mobile picking racks

1. Put a mobile picking rack on its in-use location and teach all the containers in the picking rack.
2. Repeat this for all mobile picking racks.
3. Create a variable for each picking rack.
4. Set the **Enabled** property for each container to **Based on variable**.
5. In the **Based on variable** property, select the variable you created for the picking rack the container belongs to.
6. In your assembly process, each time you exchange picking racks, set the value of the variable for the active picking rack to ON and the value of the variable for all other picking racks to OFF.

4.12 CAD/CAM detections (Validate and Vision license only)

4.12.1 About CAD/CAM detections

CAD/CAM detections allow to create a detection where the size and location in 3D space is based on a CAD model.

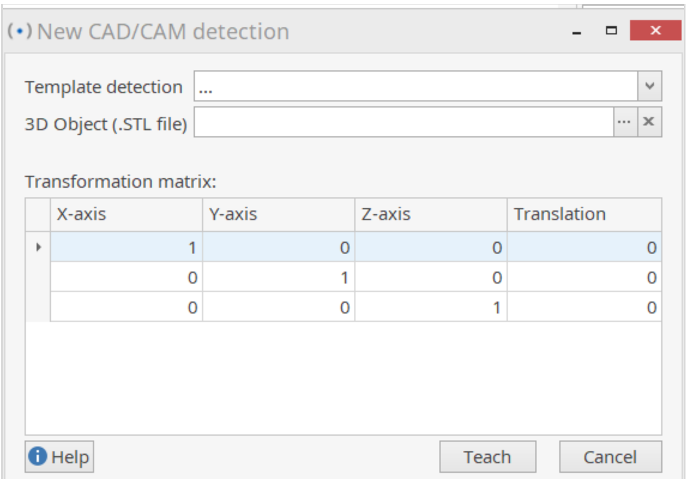
CAD/CAM detections are just like other detections (objects, tools …), but their location in 3D space is provided by both a 3D model (.STL file) and a transformation matrix that positions the CAD model in space. Such a transformation is typically provided by the CAD/CAM system. All other properties (such as which sensor to use) are taken from the template.

4.12.2 Adding CAD/CAM detections

To add a CAD/CAM detection, execute the steps below:

1. Navigate to the **Detections** tab.
2. Click the **Add** button and select **Object.**
3. Teach the object as described in section **Teach an object, container, tool or virtual button** on page 66.
4. Click the **Add** button and select **CAD/CAM Detection**.

   The **New CAD/CAM Detection** window appears.



5. Provide the input below:

Field	Description
Template detection	Select the object you have taught in step 3. All properties of the CAD/CAM detection will be identical to the one selected as template, except its location in 3D space. Such properties include which sensor to use, fill rate thresholds, whether it is RGB, depth or IR etc.
3D object	The 3D object, which is provided as an STL file (Standard Triangle Language). STL is a simple and commonly used file format, to which any CAD program can export. To select an STL file, click the 3 dots. In the **Please select an image** window, select **From File** in the **New** menu.

Field	Description
Transformation matrix	Specifies how the 3D object is positioned in 3D space, relative to the coordinate system that was defined during calibration. Such a matrix is usually provided by the CAD/CAM system, and allows to translate, rotate and scale the object as needed. The first three columns specify the XYZ-coordinates of where the X-axis, Y-axis and Z-axis of the 3D object should end up respectively. The last column specifies the translation, i.e. where the origin of the 3D object is moved to. The default values specify an identity matrix, which means the 3D object is used as-is, without transforming it.

6. Click **Teach**.

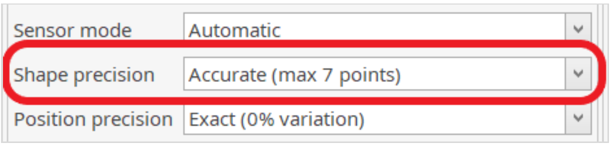
Result

Your detection is updated with the information from the STL file.

4.12.3 Example 1: no transformation

Template detection

Create a detection of type **Object**, and use all default settings, except for shape precision. Use a better precision, so the detection will accurately follow the contours of our 3D object. This is not needed, but clarifies the orientation of the 3D object in below screenshots.



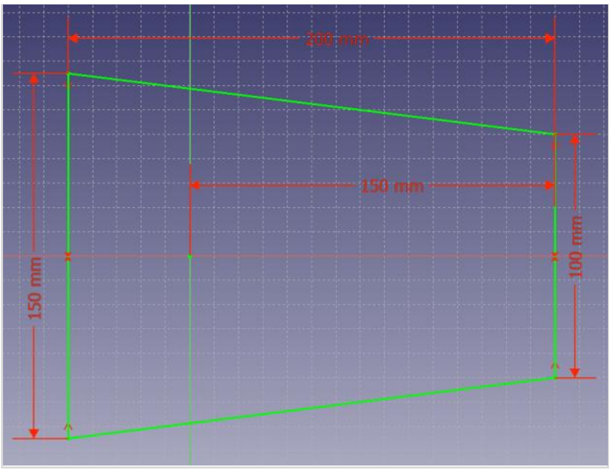
Teach the object as described in section **Teach an object, container, tool or virtual button** on page 66.

3D object

Use the below CAD model as input it can be found here:

C:\Program Files\Arkite\Arkite Workstation\Help\CAD.

The origin is in the middle of the shape, the red X-axis points right and the green Y-axis points up.



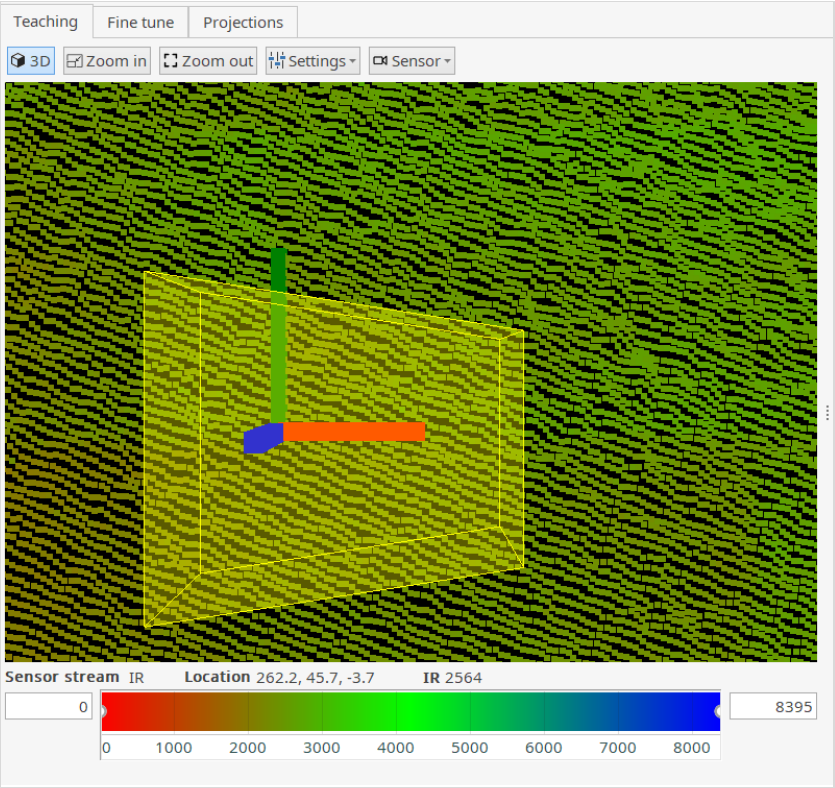
Transformation

Just use the default identify transformation.

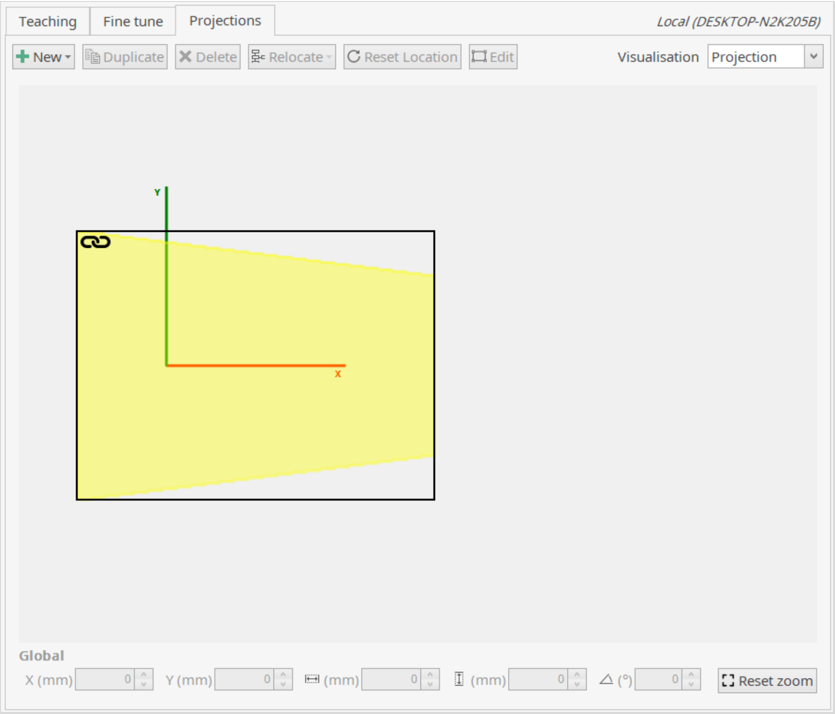
Result

This is what you see in the **Detections** tab. Notice how the axis origin specified during calibration (XYZ = RGB) aligns with the origin of the 3D object.

This is what you see in **Teaching** subtab.



This is what you see in **Projector** subtab.



4.12.4 Example 2: rotate and translate

Inputs

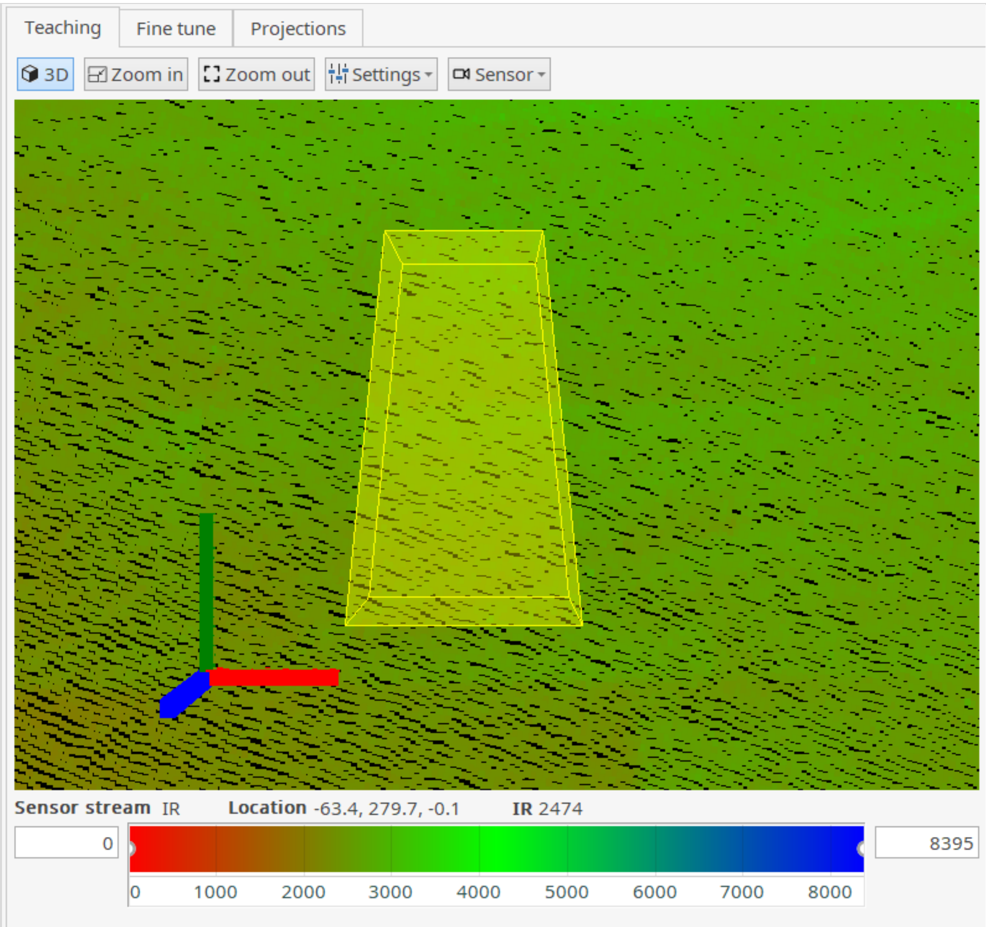
The same inputs are used as for the previous example, but this time the 3D object is rotated 90 degrees counterclockwise in the XY plane, and translated with (200, 100, 0) for XYZ. So the transformation matrix becomes:

Transformation matrix:

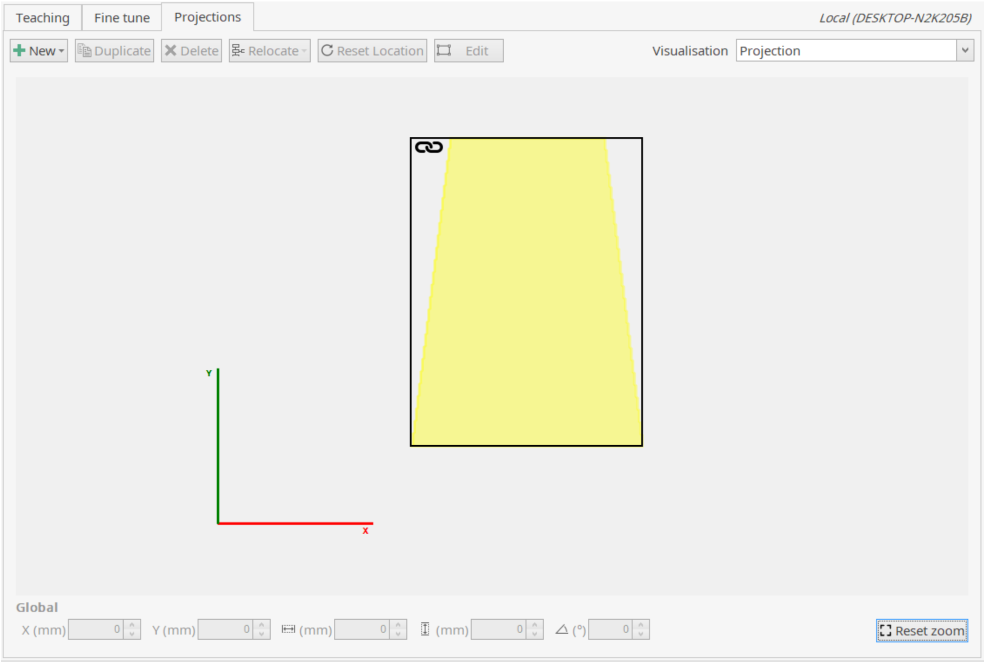
X-axis	Y-axis	Z-axis	Translation
0	-1	0	200
1	0	0	100
0	0	1	0

Result

This is what you see in **Teaching** subtab. Notice how the object is rotated, and it is translated 200mm to the right and 100mm upwards.



This is what you see in the **Projector** subtab:



5. Processes

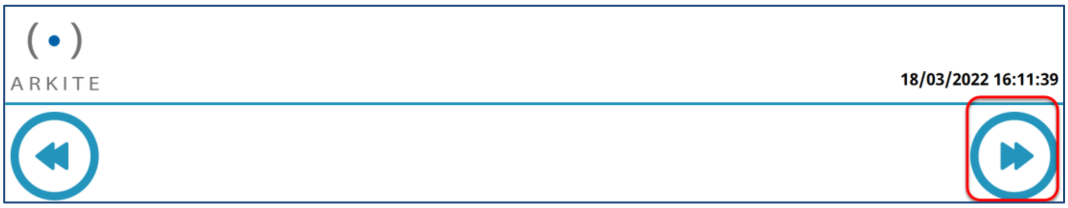
5.1 About processes and process steps

A **process** is the translation of your assembly process or your standard operating procedures, based upon the detections that you have added in your project.

A process consists of a number of **process steps**, that are executed one after another once the process has started. It's also possible to configure the process in such a way that a number of process steps are repeated or skipped.

When using a **Validate** or **Vision** license (= with Arkite 3D sensor), the system will verify if a process step has been executed and then automatically go to the next step.

When using a **Guide** license (= without sensor), the operator will always have to confirm a step has been executed by pressing the **Next** button on the touch screen, a hardware button or a foot pedal.



Apart from the assembly process as such, processes can also be used for example to:

- Log data
- Send data to a communication module
- Initialize parameters when a project is started
- Configure the behavior of a touch screen
- ...

5.2 Adding a new process

What?

When you create a project, by default one process called Product Assembly will be available that is started when the project is loaded. You can use this as your main process and start configuring you assembly steps there.

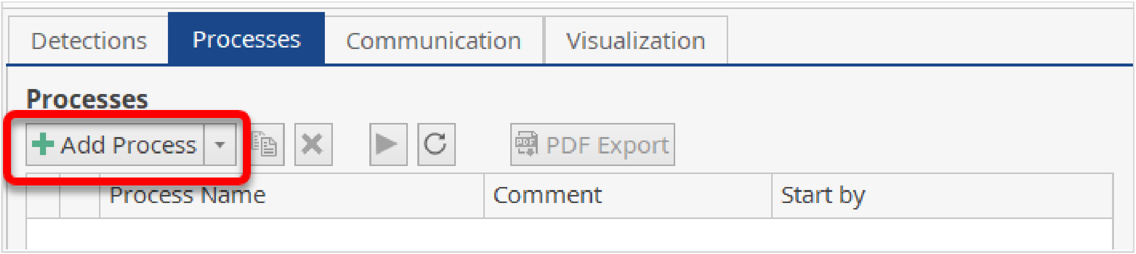
> ***Note***
>
> *For more information, refer to section **5.4 Configuring process steps** on page 104.*

You may also need more processes, for example to configure interaction with a communication module. In the section below, it is explained how to add a new process.

Steps

To add a new process, execute the steps below:

1. On the **Processes** tab, click **Add Process**.

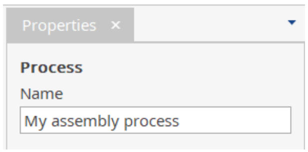


The process is added to the list of processes.

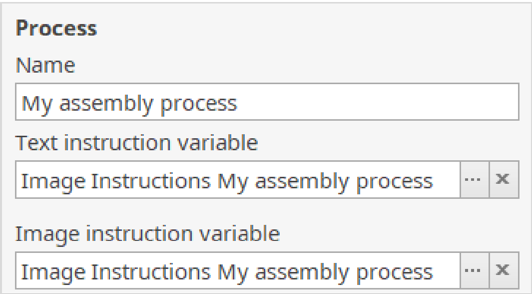
Note

*You can also add an API controlled process by clicking the arrow next to the **Add Process** button and selecting **New API controlled process**. For more information on working with the Arkite API, refer to chapter **17 The Arkite API** on 250.*

2. In the **Properties** pane, update the name of your process to something meaningful.



3. By default, one text instructions and image instructions variable is generated for all instructions for all processes. If you want to define a specific variable for the process you add, you can change the variable here.

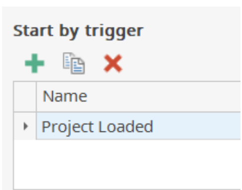


Note

*When you define specific variables, you will also have to add a specific text and image visualization to your process that displays the state of those variables. For more information, refer to section **5.5.3 Adding visualizations** on page 127.*

4. By default, a process is started when the project is loaded. Optionally, update the trigger(s) by which the process is started:

   - Under **Start by trigger**, click the button to add a trigger.



- Select the appropriate trigger.
- Configure the trigger properties.
- To remove a trigger, select the trigger and click the ✖ button.

Note

In the next section, you can find an overview of all process triggers and their properties.

Result

Your process is set up now. You can start configuring process steps to build the contents of your process.

5.3 Overview of process start triggers

5.3.1 Alarm Clock

About the Alarm Clock trigger

The **Alarm Clock** trigger allows you to start a process every day at a specified point in time. This point in time can be defined as a static value or as a variable.

Note

If there is no variable selected or if the format is incorrect, the time will be 00:00:00 by default.

Properties

Property	Description
Name	The name for the trigger. ***Note*** *Next to the **Name** label, the object ID for the trigger is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Use variables for time	Check this checkbox to use the value of a variable rather than a fixed time.
Time	The time at which the process should start.
Variable for time	The variable in the time at which the process should start is stored.

5.3.2 Project Loaded

About the Project Loaded trigger

The **Project Loaded** trigger allows you to start a process as soon as your project is loaded. This is the default trigger for a process.

Properties

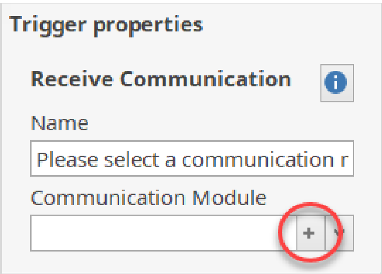
Property	Description
Name	The name for the trigger. ***Note*** *Next to the **Name** label, the object ID for the trigger is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Disable	Check this checkbox to temporarily disable the trigger, e.g. while you are still configuring your project.

5.3.3 Receive Communication

About the Receive Communication trigger

The **Receive Communication** trigger allows you to start a process within your project when data is received from an external device, such as a PLC, a smart tool, a database, a barcode scanner or another workstation.

The communication is done through a communication module. Before you can use a receive communication trigger, you first need to add and configure the appropriate communication module. Click the **+** icon in the **Communication** drop-down list.



Note

*For more information regarding the next steps, refer to the section **6.3.3 Adding a communication module** on page 178.*

Common properties

For every **Receive Communication** trigger, the properties below need to be configured:

Property	Description
Name	Name for your **Receive Communication** trigger. By default, the name of the communication module you have selected is used, but you can change it. ***Note*** *Next to the **Name** label, the object ID for the trigger is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Comment	Optionally, write a few words of explanation regarding the use of the **Receive Communication** trigger.
Communication Module	Communication module that is used to receive the communication.

The other properties depend on the communication module you have selected for communication. They are listed below for every available communication module.

Inter-Workstation

Property	Description
Variables	When the variable(s) you indicate here are received from the other workstation, the process will be started. Check the **All Variables** checkbox to start the process when receiving any variable.

Modbus

Property	Description
Variable	The variable that will be used to store the received messages.
Address Type	The object type of the read action. It is possible to read: – **Discrete inputs:** read-only entity with a size of 1 bit (ON/OFF) – **Coils**: read-write entity with a size of 1 bit (ON/OFF) – **Input registers:** read-only entity with a size of 16 bits (0-65 535) – **Holding registers:** read-write entity with a size of 16 bits (0-65 535)
Address	The address of the object from which will be read. The process will be started when a message is received from the address specified here, with the correct type.

Open Protocol

Property	Description
MID to filter on	The MID (which is character 5 to 8 of the message header) to filter the messages on. If this field is filled in, only messages with a matching MID will be captured and the process will only be started when the matching MID is received. If this field is left empty, all messages will be captured. The process will started whenever a message is received.
Variable for message	The variable that will be used to store the received messages.

5.3.4 Timer

About the timer trigger

The **Timer** trigger allows you to start a process with a certain interval.

Example

With a combination of a Timer trigger and a Counter actuator you can create a clock.

Properties

Property	Description
Name	The name for the trigger. ***Note*** *Next to the **Name** label, the object ID for the trigger is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Milliseconds in between sparks	The interval with which the process should start.

5.3.5 Watchdog

About the Watchdog trigger

The **Watchdog** trigger allows you to let a process run as long as no errors in hardware of software are detected. The watchdog will check the system with a certain interval and flag possible malfunctions.

> ***Note***
>
> *Combined with an ethernet communication module and external hardware like a light or buzzer, it can be used to visualize the state of your system.*

Properties

Property	Description
Name	The name for the trigger. ***Note*** *Next to the **Name** label, the object ID for the trigger is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Milliseconds between ticks	The interval with which the watchdog checks the system's hardware and software features.
Minimum FPS	The minimum frames per second the sensor should have.
Variable for message	The variable that will store the state of the watchdog (OK or FAILURE). The process will only start when the state of this variable is OK.
Variable for failure	The variable that will store the watchdog's error message when its state is FAILURE

5.3.6 Variable Changed

About the Variable Changed trigger

The **Variable Changed** trigger allows you to start a process as soon as one or more variables reach a certain state or are changed.

Properties

Property	Description
Name	The name for the trigger. ***Note*** *Next to the **Name** label, the object ID for the trigger is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Any change	Check this checkbox to start the process at any change in state. In this case, there is no desired state.
Compare with	Check this checkbox to use the state of another variable as the desired state.
Variables	Define the variable(s) you want to monitor for state change.
Comparison	Select a compare operator
Other variable	Define the variable you want to compare the variable you monitor with.
Desired state	Define which state the variable(s) should reach to start the process

Property	Description
Type	– Select AND if all variables need to reach the desired state. – Select OR if only one of the variables need to reach the desired state. – Select Probing to use the probing algorithm. More information about can be found below this table.
State to set (probing)	The probing algorithm will select 1 variable. The state of this variable will be set to the value in this property for 1 ms. After that, it returns to its previous state. You can use this change to make a change to another variable.
Premature Probing Spark	If not checked, the probing algorithm will wait until all variables have returned to their default values. Then it will select the variable that has been in the desired state for the longest time. If checked, the probing algorithm will select a variable when it has reached desired state for a significantly longer time than any other variable.
Premature Spark Time	Set the minimum time in milliseconds before the probing algorithm selects a variable.
Premature Spark Fraction	Define how much longer a variable needs to be in the desired state than any other variable for the probing algorithm to select it.

Probing

In case you select probing, there are two possibilities for the start of your process:

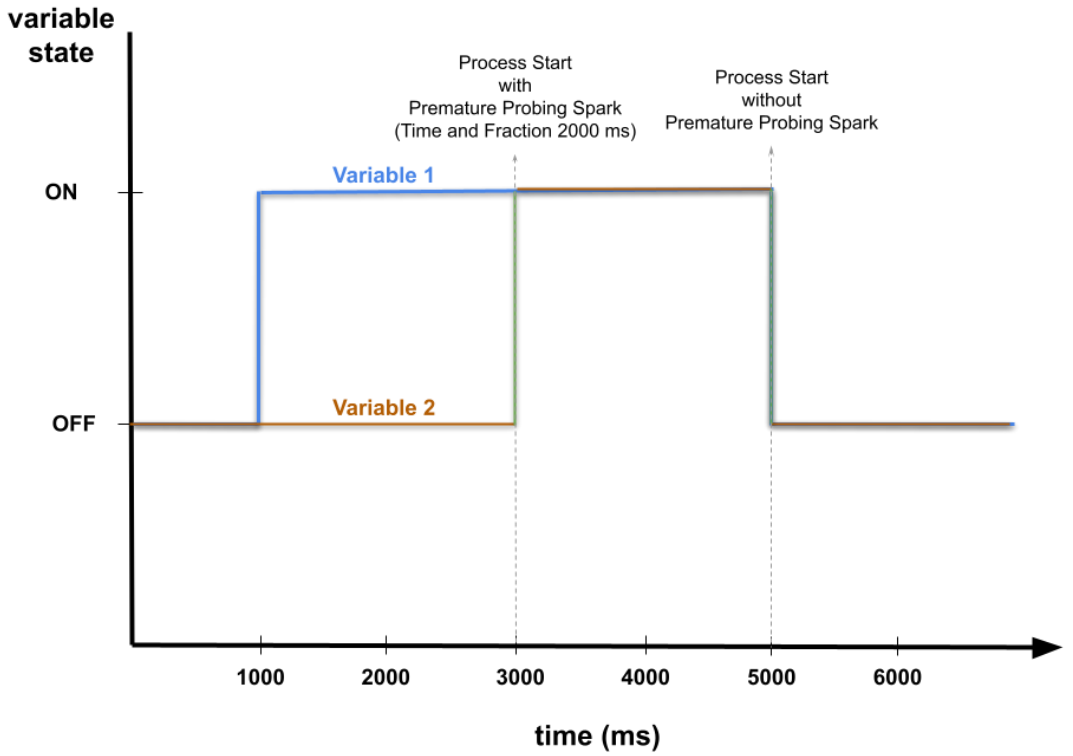
- When **Premature Probing Spark** is <u>not</u> checked, your process will be started when all variables have reached the desired state at one point ***and*** returned to their default state afterwards.
- When **Premature Probing Spark** is checked, your process will be started:
  - When the time set in the **Premature Spark Time** property has passed, AND
  - When one variable was in the desired state significantly longer than any of the other variables. To define what is significantly longer, the time set in the **Premature Spark Fraction** is taken into account.

In both cases, the variable that was in the desired state for the longest time is changed to the value defined in the property **State to Set (probing)** for 1 ms. After that, it returns to its previous state.

Example

Suppose we have:

- 2 variables with default state OFF.
- Probing is selected, with desired state ON and state to set PROBING
- Variable 1 changes to state ON after 1 second
- Variable 2 changes to state ON after 4 seconds
- Both variables go back to their default state OFF after 5 seconds.



In case **Premature Probing Spark** is not checked, the behaviour will be as in the table below.

The process is started after 5 seconds, when both variables are back in their default state.

The probing algorithm selects variable 1 because it has been in state ON for the longest time and changes its state to PROBING for 1 ms.

Time (ms)	Variable 1	Variable 2	Process status
0	OFF	OFF	Not started
1000	ON	OFF	Not started
2000	ON	OFF	Not started
3000	ON	OFF	Not started
4000	ON	ON	Not started
5000	OFF	OFF	Started
5001	PROBING	OFF	Running
5002	OFF	OFF	Running
6000	OFF	OFF	Running

In case **Premature Probing Spark** is checked with both **Premature Spark Time** and **Premature Spark Fraction** set at 2000 ms, the behaviour will be as in the table below.

The process is started after 3 seconds, when variable 1 has been in state ON for 2 seconds longer than variable 2.

The probing algorithm selects variable 1 and changes its state to PROBING for 1 ms.

Time (ms)	Variable 1	Variable 2	Process status
0	OFF	OFF	Not started
1000	ON	OFF	Not started
2000	ON	OFF	Not started
3000	ON	OFF	Started
3001	PROBING	OFF	Running
3002	ON	OFF	Running
4000	ON	ON	Running
5000	OFF	OFF	Running
6000	OFF	OFF	Running

5.4 Configuring process steps

5.4.1 Process step types

This section contains an overview of all types of process steps that are available to you to build your process. They are divided in the categories below:

- **General steps**: some basic, often used process steps.
- **Detections**: process steps related to the detections you have added
- **Materials:** process steps related to the materials you have configured.
- **Screen buttons:** process steps related to the buttons you have added to the touch screen.
- **Communication:** process steps related to the communication modules you have configured.
- **Product:** process steps related to the logging of the start, completion or scrap of an assembly.
- **Extra**: a number of advanced, less often used process steps.

General steps

The table below gives an overview of all process step types that are available in the **General steps** subtab:

Process step type	Description
Composite step	A step that can contain a number of sub steps. It allows you to control the order in which sub steps need to be executed or to work with alternative steps. ***Note*** *For more information, refer to section* ***5.4.3 Organizing your process steps*** *on page 112.*
Task step	A number of steps coming from a task. Tasks allow you to re-use the same steps across processes. ***Note*** *For more information, refer to section* ***5.8 Working with tasks*** *on page 152.*

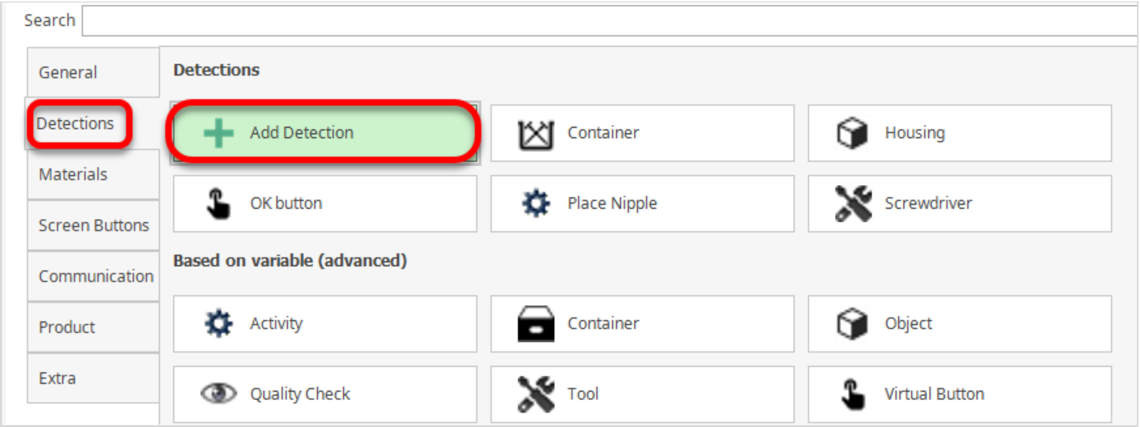
Process step type	Description
Variable changed	Check the value of a particular variable. Depending on the specific settings, the process will continue if: – the variable has a particular value – the variable does not have a particular value – the variable's value matches another variable's value – the variable's value does not match another variable's value – the variable's value is changed
Delay	Set a certain time in milliseconds for the process to wait.
Work Instruction	Provide additional text and/or image instructions. ***Note*** *The process will never automatically continue after a* ***Work Instruction*** *step, Therefore you always need to provide a* ***Next*** *or* ***Confirm*** *button on the screen or combine it with a* ***Process Control*** *step.*
Dummy	Introduce a placeholder for a step that you want to define later.
Set Active Variant	Change the active variant in your project ***Note*** *For more information on working with variants, refer to section* ***5.6 Working with variants*** *on page 139.*
Switch User	Logs out the current user. You can indicate a new user to sign in with, or show a login screen.
Switch Language	Sets the language in which instructions are displayed. ***Note*** *By default, instructions are displayed in the user language of the user that is logged in. If no instructions are available in that language, the system will look for a similar language. For example, if the user language is en-GB, en-US will be used. If no similar language is available, the first instruction language in the project will be used.*
Set Variable	Update the value of a variable or a list of variables. In the **State to Set** field, enter the value you want to set for the selected variable(s). To insert dynamic content, click the icon at the location where you want to include the content and select the variable containing the dynamic content. Alternatively, you can also type the variable name between double braces, i.e. *{{Variable name}}.* You can also use a variable within a variable. The variable you want to update can also be dynamic: by selecting the **By Name** option you can provide a variable name with fixed and dynamic parts. This allows you to update variables in a loop. To do so, embed the step in a composite step that is repeated for a number of elements. ***Example*** *If you provide the name* ***Orientation_x{{index}}****, the variable that is set will depend on the value of the* ***index*** *variable. If the value of the* ***index*** *variable is 01 at the time of execution of the step, the variable* ***Orientation_x01*** *will be set.*

Process step type	Description
Process Control	Control a particular process. That process can also be based on the value of a variable. ***Note*** *If the variable contains a value that does not match with a process name, the **Process Control** step will not be executed and an error will be added in the error tray.* You can choose to: – Reset a process – Pause a process – Pause and reset a process – Play a (paused) process – Restart a process, i.e. reset and play it – Go to the next step in a process – Go to the previous step in a process – Go to a specific step in a process. That step can also be based on the value of a variable. ***Note*** *If the variable contains a value that does not match with a step name, the **Process Control** step will not be executed and an error will be added in the error tray.* *If the variable contains the name of the **Process Control** step itself, the **Process Control** step will not be executed.* *If the variable contains a value that matches with multiple step names, the first step with a matching step name will be activated.*
Variable Logger	Log the current state of one or more variables.
Layer Visualization	Show or hide layers on a particular display.
Open/Close Display	Open or close a particular display.
Counter	Increment or decrement the value of a variable by 1
Project Loader	Restart the current project or load another project
Resetter	Reset a variable or entity to its default state.

Detections

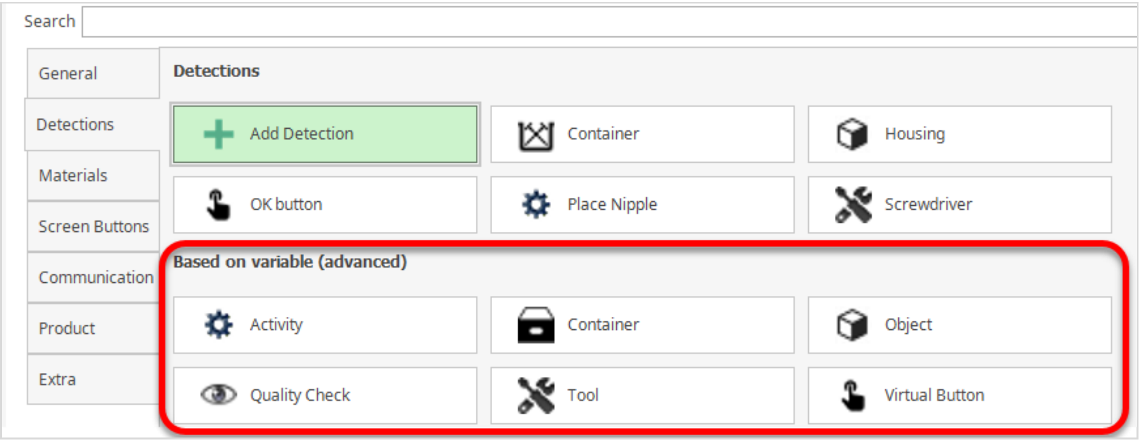
At a certain point in your process, you can integrate a step in which an action is done with a detection.

However, before you can do so, you first need to add the detection. Click the **Add Detection** button in the **Detections** subtab.



For more information regarding the next steps, refer to section **4.4 Adding detections** on page 60.

You can also add a step in which the detection that is used depends on the value of a variable, defined in the **Variable for detection name** property. If this variable contains a value that does not match with a detection in the project, or with a detection that is not taught, the project will be paused. An error will be shown in the error tray and logged in the variable configured in the **Variable for error message** property.



The actions that are available to be used in a process step depend on the detection type. The table below provides you with an overview of available actions per detection type.

Detection type	Available actions
Object	– **Place:** put the object at the location where it should be. – **Take:** remove the object from the location where it should be.
Tool	– **Place:** put the tool at the location where it should be. – **Take:** remove the tool from the location where it should be.
Container	– **Place container:** put the container at the location where it should be. – **Take container:** remove the container from the location where it should be. – **Pick:** take material from the container ***Note*** *If you are adding a pick step, it is often better to use the material from the* ***Materials*** *subtab, rather than the container. If at some point, the material is linked to a different container, your pick step will be automatically updated.*
Activity	**Activity**: perform the activity.
Virtual button (Validate and Vision license only)	**Press**: press the virtual button.
Quality Check (Vision license only)	**Check:** perform the quality check.

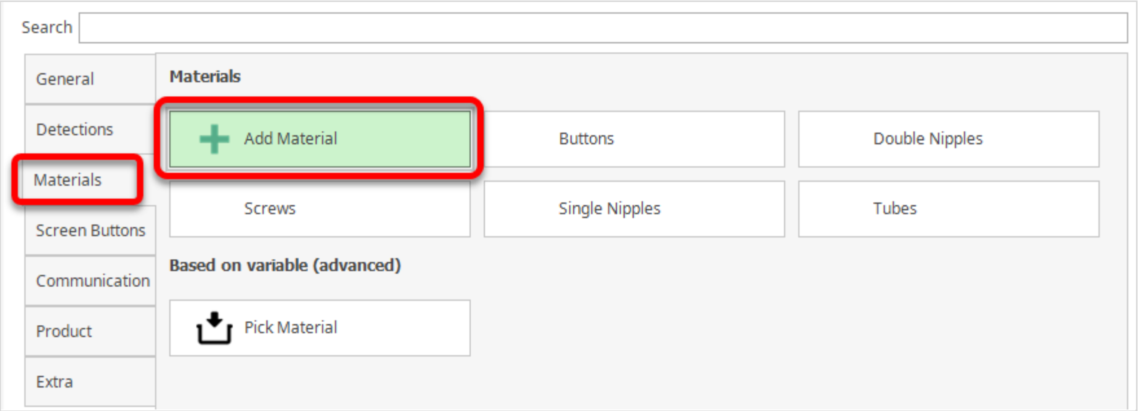
Note

When a detection is not taught, the step cannot be completed: the action that is required to trigger the detection is not available. If teaching the detection is not possible (when using the system with a ***Guide*** *license), you always need to provide a* ***Next*** *or* ***Confirm*** *button on the screen or combine it with a* ***Process Control*** *step.*

Materials

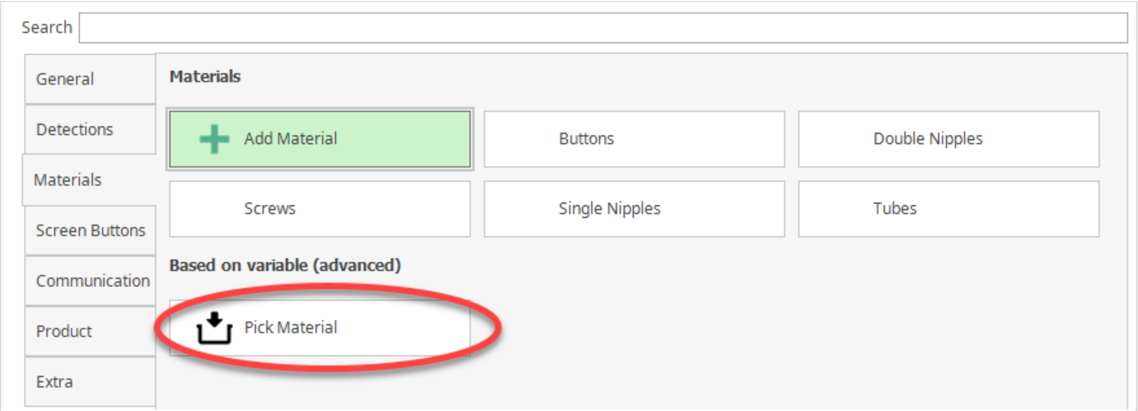
At a certain point in your process, you can integrate a step in which a material is picked from a container.

However, before you can do so, you first need to add the material. Click the **Add Material** button in the **Materials** subtab.



For more information regarding the next steps, refer to section **4.7 Managing materials** on page 82.

You can also add a step in which the material that is picked depends on the value of a variable, defined in the **Variable for material name** property. If this variable contains a value that does not match with a material in the project, contains a material that is not linked to a container or is linked to a container that is not taught, the project will be paused. An error will be shown in the error tray and logged in the variable configured in the **Variable for error message** property.



Note

When there is no container defined for the material, or if the defined container is not taught, the step cannot be completed: the action that is required to trigger the detection is not available. If teaching the container is not possible (when using the system with a ***Guide*** *license), you always need to provide a* ***Next*** *or* ***Confirm*** *button on the screen or combine it with a* ***Process Control*** *step.*

Screen buttons

At a certain point in your process, you can integrate a step in which a screen button on the touch screen is pressed.

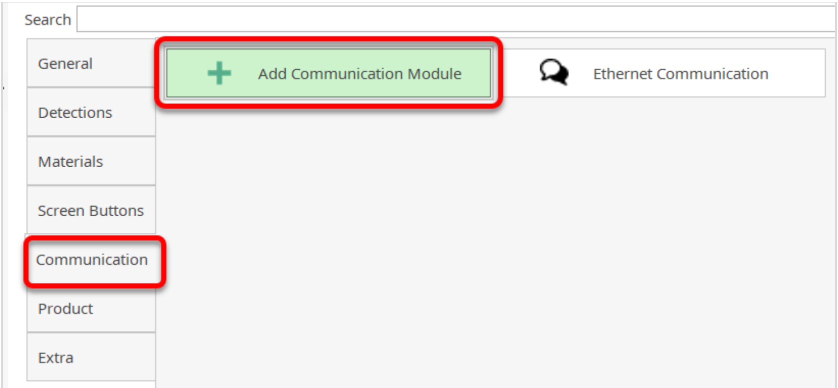
However, before you can do so, you first need to add the button on a **Screen** display. Until you have done this, the **Screen buttons** subtab in the overview of process steps will be empty.

For more information, refer to section **5.5.3 Adding visualizations** on page 127.

Communication

At a certain point in your process, you can integrate a step in which a communication is sent.

However, before you can do so, you first need to configure a communication module. Click the **Add Communication Module** button in the **Communication** subtab.



For more information regarding the next steps, refer to section **6.3.3 Adding a communication module** on page 178.

Product

The **Product** subtab contains steps to register the start and completion and scrapping of product assemblies. They are used for building the **Product Output** and **Cycle Times** dashboards.

> ***Note***
>
> *For more information about consulting log tables, refer to section **12.2 Consulting log tables in the Arkite software** on page 220.*
>
> *For more information about dashboarding, refer to chapter **11 Dashboarding** on page 215.*

The table below gives an overview of all process step types that are available in the **Product** subtab:

Process step type	Description
Log Product Start	Registers the start of the assembly of a product in the PRODUCT_LOG table. By default, the product name is the name of the active variant. It is also possible to define the product name in the step properties, or use the value of another variable.
Log Product Complete	Registers the completion of the assembly of a product in the PRODUCT_LOG table. The **Product Name** and **Product Identifier** properties are taken over from the preceding **Log Product Start** process step.
Log Product Scrap	Registers the scrapping of the assembly of a product in the PRODUCT_LOG table. Typically, it is used when an error occurs during the assembly. Use it in a separate process, in combination with a **Process Control** process step that resets the **Product Assembly** process. The **Product Name** and **Product Identifier** properties are taken over from the preceding **Log Product Start** process step.

Note

*By default, your **Product Assembly** process contains a **Log Product Start** and **Log Product Complete** step. You can add more **Product** process steps if required, for example for multiple product variants. You should make sure there is a **Log Product Complete** step for every **Log Product Start** step. If not, the system will not be able to correctly calculate assembly times and generate dashboards. A warning message will be shown in the corresponding dashboards.*

Extra

The table below gives an overview of all process step types that are available in the **Extra** subtab:

Process step type	Description
Calculator	Allows you to calculate a formula with fixed numbers and/or dynamic content based on the value of one or more variables. The operators below are supported: + : addition of numbers - : subtraction of numbers * : multiplication of numbers / : division of numbers: ▪ If all operands of the division are integers, the result will be an integer too, e.g. 7/2=3 ▪ If at least one of the operands in the division is a decimal number, the result will be a floating division, e.g. 7/2.0=3.5 %: remainder after dividing numbers ***Note*** *Always use . as the decimal separator to separate the integer part from the fractional part of a number.* *To define the order of several operations, brackets () can be used.*
Split String By Index	Extracts a fragment of a string in a variable based on: – **index:** the position of the first character of the string – **length:** the number of characters to extract, starting from the index
Split String By Argument	Extracts a fragment of a string in a variable based on: – **split symbol:** a character used to split the string – **index:** the number of occurrences of the split symbol after which you want to extract the fragment Use the **Variable** property for the string that needs to be split, the **Variable for Result** property for the extracted fragment and the **Variable for remainder** property for what is left of the string after extracting the fragment you needed. ***Example*** *If the string you want to split is **"Arkite;is;a;great;system;for;guiding;operators"** and you want to retain only the word **great**, then you can use ; as a split symbol and 4 as index.* *In that case, the variable indicated in the **Variable for remainder** property will contain what is left from the string, i.e. **Arkite;is;a;system;for;guiding;operators"***
Restore Point	Stores the current project state or loads a stored project state. The project state (i.e. steps, sequences and variable states) are stored in an .hrp file.
System Command	Issues one of the system commands below: – **Shut down:** shut down your workstation – **Reboot:** reboot your workstation – **Restart Arkite:** close and restart the Arkite software – **Project:** open the **Menu** tab and the **Projects** subtab

Process step type	Description
Printscreen	Saves an image of the current screen. ***Warning*** *Don't use this with a sensor or timer, because that will take a lot of disk space.*
String Formatter	Compiles a text string containing fixed text and/or dynamic text based on the value of one or more variables. To insert dynamic text, click the icon at the location where you want to include the text and select the variable containing the dynamic text. Alternatively, you can also type the variable name between double braces, i.e. *{{Variable name}}*. You can also use a variable within a variable. For example, if the value of a variable **Index** is 01, then *{{Variable{{Index}}}}* will resolve into the value of the variable **Variable01**. ***Tip*** *For long text strings, click the icon to enlarge the* ***String to format*** *text box.*
String Parser	Searches a JSON or XML string for the names of specified variables and stores the corresponding values of the first match as the current state of those variables.
File Mover	Move an existing file to a new location or changes the name of that file
Frames Recorder (Validate and Vision license only)	Starts or stops the recording of frames. The behavior is similar to clicking the **Record** button in the ribbon.
Snapshot	Takes a snapshot from the selected feed and stores it as a *.jpeg
Sound	Plays a sound file in .wav format.

5.4.2 Adding process steps

What?

By default, your Product Assembly process contains two process steps: a **Log Product Start** and **Log Product Complete** step to register the start and completion of an assembly. The other steps of your assembly must be added.

Steps

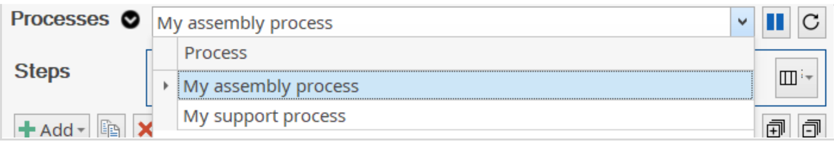
To add a new process step, execute the steps below:

1. Navigate to the **Processes** tab.
2. Under **Processes**, select the process in which you want to add a new process step.

   Note

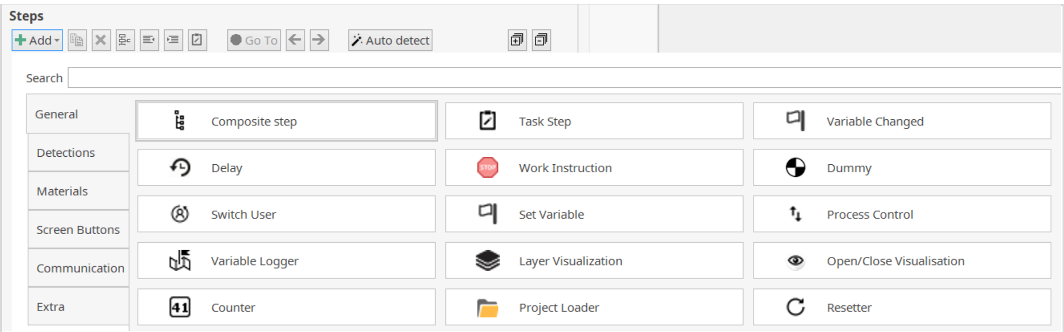
   For a better overview and more space when adding process steps, you can collapse the ***Processes*** *section by clicking the button.*

   You can still select the active process in the dropdown list.

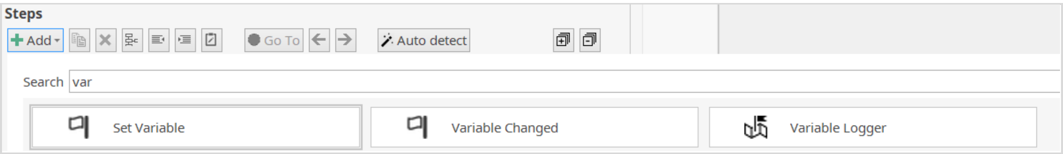
   

   To expand the ***Processes*** *section again, click the button.*

3. Under **Steps**, click the **Add** button and select the process type you want to add.



You can also enter the process type in the **Search** field to find it.



Result

The process step is added to the list of process steps.

5.4.3 Organizing your process steps

About organizing your process steps

By default, your process will be executed in the order the steps appear in your process. For a better overview, you can group individual steps into composite steps. That also allows you to repeat one or more steps, or indicate steps as optional, either by working with variants, with conditions or with alternative or arbitrary steps.

Note

*For a better overview and more space when organizing process steps, you can collapse the **Processes** section by clicking the button.*

You can still select the active process in the dropdown list.

*To expand the **Processes** section again, click the button.*

Composite steps

A composite step is a step that consists of two or more child steps. These child steps can be individual steps or other composite steps, that in their turn also consist of child steps.

In the **Properties** pane for a composite step, you can define in which order and how often the child steps need to be executed.

Property	Description
Name	The name for the process step.

Property	Description
	Note *Next to the **Name** label, the object ID for the process step is shown. You can copy it to use it when working with the Arkite API.*
Child step order	Define in which order the child steps are executed: – **Sequential order (icon):** The child steps are executed in the order in which they appear. This is the default option. – **Any order (icon - Validate and Vision licenses only):** It does not matter in which order the child steps are executed. However, all child steps must be executed for the process to continue to the next step. – **One arbitrary step (icon):** Only one of the child steps should be executed for the process to continue to the next step. – **All steps simultaneously (icon - Validate and Vision licenses only):** All steps should be executed at exactly the same time for the process to continue to the next step. For example, you can use this type of step to check if all tools are present when you start the assembly, or if you want an operator to press two virtual buttons at the same time to confirm an action, as a safety measure. ***Note*** *Do not use the **Any order and All steps simultaneously** options when using a **Guide** license, because in that case there is no way to verify which child steps are executed by the operator. When running a project that contains such steps on a system without sensor, these steps will be treated by the system as sequential steps.*
Repetition	Indicate if the composite step should be repeated: – **None:** The composite step is executed only once. After execution of the last child step, the process continues with the step after the composite step. – **Repeat:** The step is repeated in an endless loop. After the execution of the last child step, the first child step is executed again. ***Note*** *This feature should be used in combination with a condition: the composite step is repeated as long as the condition is valid. For more information about working with conditions, refer to section **5.7 Working with conditions** on page 144.* – **Repeat number of times:** The step is repeated a specified number of times (see below). – **Repeat for each element:** The step is repeated for all elements in a specified list (see below).
Number of repetitions	If you have indicated the step should be repeated for a specified number of times, enter the number of repetitions in this field. You can also specify a variable that contains the number of repetitions. To do so, click the icon and select the variable containing the number of repetitions. Alternatively, you can also type the variable name between double braces, i.e. {{Variable name}}.
List of elements to loop over	If you have indicated the step should be repeated for all elements in a specified list, enter the list here. You can also specify a variable that contains the list. To do so, click the icon and select the variable containing the list. Alternatively, you can also type the variable name between double braces, i.e. {{Variable name}}.
List type	If you have indicated the step should be repeated for all elements in a specified list, enter the list type here:

Property	Description
	– Json Array – List with separator
Separator	If you have indicated the step should be repeated for all elements in a specified list of type **List with separator**, enter the separator here.
Variable for current element	If you have indicated the step should be repeated for all elements in a specified list, this variable will be updated with the current element.
Variable for count	If you have indicated the step should be repeated for all elements in a specified list, this variable will be updated with the total number of elements in the list.
Variable for index	If you have indicated the step should be repeated for a specified number of times or for all elements in a specified list, this variable will be updated with the current iteration.

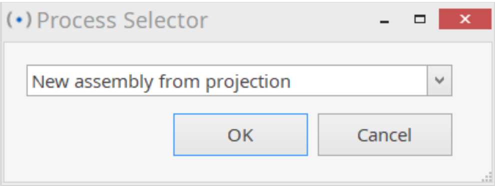
The root step

Your complete process is organized as a number of child steps of one global process step. This step that contains the whole process is called the root step. It has the same features as a regular composite step.

How to organize your process steps

A number of buttons are available to help you organize your process steps:

- Select a process step and click to duplicate it.
- Select a process step and click to delete it.
- Select a process step and drag it with your mouse to change its order.
- Select a process step and click to move it to another process.
  Select the process to which you want to move the process step and click **OK**.



- Create composite steps:
  - Click **New** and select **Composite Step**.
    A composite step is created. Any new process step you add while this group is selected will be added as a child step of this step. You can also drag existing process steps into the composite.
  - Select one or more process steps and click the button. This way, a composite step can also be created.
- Click the button to increase the level of detail you see.
- Click the button to decrease the level of detail you see.

5.4.4 Testing your process and process steps

You can test your process by selecting the process and clicking the ▶ button, or by clicking the **Run** button in the ribbon.

The circular icons for every process step indicate the progress of your process:

Icon	Description
○	The step has not been started yet.
● ◑	The step is currently active (Run mode). For composite steps, the part of the circle that is filled indicates the percentage of the child steps that is executed.
	The step is currently active, but is suspended because the project is not running (Edit mode). For composite steps, the part of the circle that is filled indicates the percentage of the child steps that is executed.
●	The step has been completed.
	The step will be ignored due to variant and condition logic.
□	The step is currently blocking the process – in its current state, the process will never continue to the next step. This can have several causes: – A detection is not taught (**Detection** step) – A detection is disabled (**Detection** step) – in this case, additionally a ▮▮ icon will also be shown with the step. – There is no container defined for a material (**Material** step) – There is no container taught for a material (**Material** step) – There is no trigger defined for the step (default behaviour for **Work Instruction** and **Dummy** steps)

Note

*On an editor, i.e. a workstation without a valid **Guide**, **Validate** or **Vision** license, you will be able to run a project for a maximum of 15 minutes. After that, it will automatically revert to **Edit** mode.*

To force the process to go to the next step, click the → button under **Steps**.

To return to the previous step, click the ← button under **Steps**.

To go to a specific step, select the step and click the Go To button.

5.5 Managing visualizations for your process steps

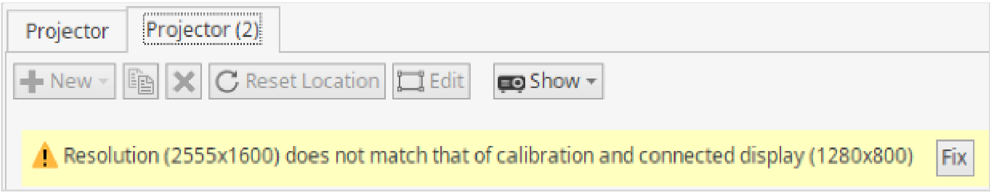
What?

After creating your process steps, you can manage the visualizations that will be displayed when those steps are active, on a connected projector or screen.

You will see a subtab for every display that is available. By default, a **Projector** subtab will alwaysbe available.

Note

- *When you have created the project before calibrating, a **Projector** display is added with the screen resolution of the device you are working on (e.g. your laptop). This may not match with the actual projector resolution, used during calibration. This will cause projections to be faulty. A message on the **Projector** subtab will warn you for this. Click **Fix** and update the **Resolution** property to the correct value in the projector's **Properties** pane.*



- *To add more displays, refer to section **6.2.3 Adding a display** on page 161.*

When creating a process, the Arkite software will automatically generate visualizations to display text and image instructions.

Additionally, you may also create more visualizations to display.

When the **Experience Level** functionality is enabled, you can add experience level buttons to a **Screen** display to allow an operator to select his experience level.

5.5.1 Visualization types

Following types of visualizations are available in the Arkite software:

- **Draw shape:**
  A geometrical figure. It can be a rectangle, an ellipse, a line or a polygon.
- **Text:**
  Visualization that allows you display a text to the user. The text can contain both fixed and dynamic parts, based on the value of a variable.
- **Image:**
  Visualization that allows you to display a fixed or dynamic image or video to the user.
- **Button:**
  Screen button. Allows the user to click it and set a variable and/or control a process in your project by doing so.

  Note

  *The functionality of a button only works when added to a **Screen** display.*
- **Counter:**
  Visualization that shows the value or spark rate of a counter.
- **Progress bar:**
  Visualization that shows the progress of a process, or a progress bar based on the state of two variables.
- **Operator Input:**
  Visualization that allows you to add an input field to a **Screen** display, in which the operator can type information. The information is stored in a variable.

- **Keyboard:**
  Visualization that displays an on-screen keyboard onto a **Screen** display. The operator can use it to enter information in an **Operator Input** visualization if no physical keyboard is available.
- **Sensor stream with detections (Validate and Vision license only):**
  Visualization that displays a specific set of detections with their current state.

5.5.2 Visualization properties

The sections below provide you with an overview of the properties that can be defined for each visualization type.

First, the properties that are available for each visualization type are described:

- Name property
- Visibility properties
- Color properties
- Experience Level properties

Next, the additional specific properties for each visualization type are described.

Name property

The **Name** property allows you to define a specific name for the visualization.

> ***Note***
>
> *Next to the **Name** label, the object ID for the visualization is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*

Visibility properties

For each visualization, the properties in the table below can be configured in the **Visibility** section:

Property	Description
Visibility	Select one of the options below from the **Visibility** dropdown list: – **On**: the visualization will always be visible – **Blink**: the visualization will blink. Use the **Blink Interval** property to define how fast the visualization will blink. – **Based on Variable**: the visibility of the visualization will be based on the current state of the variable provided in the **Variable for visibility** property.
Variable for visibility	Only available if the **Based on Variable** option is selected in the **Visibility** dropdown list. The state of this variable will determine whether the visualization will be visible: – If the state of the variable is ON, TRUE or 1, the visualization will be visible. – If the state of the variable is OFF, FALSE or 0, the visualization will be hidden. – If the state of the variable is BLINK, the visualization will blink. Use the **Blink Interval** property to define how fast the visualization will blink. – If the variable has another state than the ones mentioned above, the visualization will be visible.
Blink Interval (ms)	Only available if the **Blink** option is selected in the **Visibility** dropdown list. Interval in ms with which the visualization will blink.

Property	Description
Rotate for verify position (Vision license only)	Check this option to rotate the visualization depending on the actual position of the linked detection compared to the expected position. You can use this for example to display an arrow that indicates in which direction the detection should be moved in order to be on the expected position. Only available when **Verify Position** has been selected when teaching the detection.

Note

*When the visibility is based on a variable, the visualization will always be visible in the UI and when your project is in **Edit** mode. In **Run** mode, the visibility as specified by the variable will be used.*

Color properties

For each visualization, the properties in the table below can be configured in the **Colors** section:

Property	Icon	Description
Background Color		Set the visualization's background color.
Background Border Color		Set the visualization's background border color.

Note

- *For some visualizations, additional color properties can be set. These are explained in the sections below.*
- *For each color property, you can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section **5.5.5 Managing visualization or DMX light colors** on page 130.*

Experience Level properties

In case the **Experience Level** functionality is enabled, a check box is available for each experience level. This way, you can indicate for which experience level(s) the visualization should be shown. By default, every visualization is shown for all experience levels.

Note

*For more information on the Experience Level functionality, refer to chapter **7 Experience levels** on page 194.*

Draw shape

A **draw shape** visualization has the properties in the table below:

Property	Icon	Description
Fill shape		If checked, the shape will be filled with the brush color. If unchecked, the shape will be transparent and the background color will be visible.

Property	Icon	Description
Brush Color		If the **Fill Shape** property is unchecked, a brush on the outside of the shape will have this color. If the **Fill Shape** property is checked, the entire shape will have this color. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section* ***5.5.5 Managing visualization or DMX light colors*** *on page 130.*
Brush Size	3	Thickness of the brush

Text

A **text** visualization has the properties in the table below:

Property	Icon	Description
Text Font Size	64	The font size of the displayed text.
Text Color	A	The font color of the text. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section* ***5.5.5 Managing visualization or DMX light colors*** *on page 130.*
Text Align		The horizontal alignment of the text. This can be either left, center or right.
Text		Provide the text you want to display, in any available language. To insert dynamic text, click the icon at the location where you want to include the text and select the variable containing the dynamic text. Alternatively, you can also type the variable name between double braces, i.e. *{{Variable name}}*. You can also use a variable within a variable. For example, if the value of a variable **Index** is 01, then *{{Variable{{Index}}}}* will resolve into the value of the variable **Variable01**. ***Tip*** *For long texts, click the icon to enlarge the text box.* ***Note*** *To add or remove languages, refer to section* ***15.2 Managing languages*** *on page 237.*
Show in PDF		If checked, the text visualization will be shown in the PDF export.

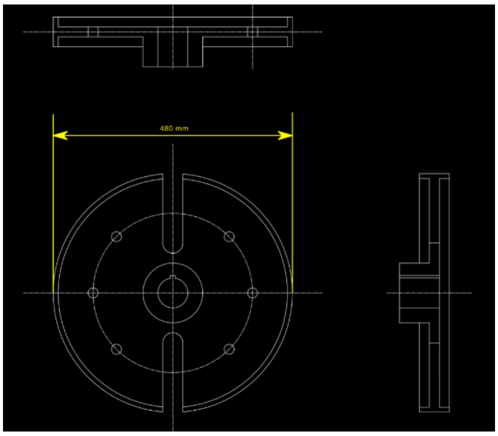
Image

An **image** visualization has the properties in the table below:

Property	Description
Image based on variable	If checked, the path of the image/video or the SVG content for the image that should be displayed is provided as the state of a variable. If unchecked, the image or video that should be displayed is provided in the **Image** property.
Image	Image or video that should be displayed

<table>
<tr><th>Property</th><th>Description</th></tr>
<tr><td>Variable for image</td><td>Specifies the variable(s) of which the state contains either:
– The SVG (Scalable Vector Graphics) content for the image to be displayed. Check the Parse variable as SVG property for the system to render the image correctly
– The path to the image or video that should be displayed. It looks for an image/video with a matching name within the project images as available from the Images button on the ribbon.
Note
If multiple variables are provided, the one that changed most recently is used.</td></tr>
<tr><td>Search on local drive</td><td>If checked, the system will search on the local drive instead of in the project images to find the image/video to display.
The state of the Variable for image should specify the file name, with or without file extension, but always without the folder name.</td></tr>
<tr><td>Search local folder</td><td>The folder in which the system will look to find the image/video to display.
It does not look into subdirectories. The folder is local in that it looks for files in the specified folder on the workstation on which the project happens to be running. So different workstations may have a folder with the same name, but it may contain different files.
Example
Typically, this feature can be used to display a snapshot that was taken in a Snapshot process step. To make things work, use the same folder you have specified in the Snapshot process step. In the Variable for image property use the same variable you have specified in the Snapshot process step.</td></tr>
</table>

Property	Description
Parse variable as SVG	If checked, you can use SVG (Scalable Vector Graphics) contents in the state of the **Variable for image**. The system will parse the variable and render the image. An example use case is where feedback from a vision system provides areas to highlight, by writing such SVG contents to a variable via the API. ***Note*** *When using SVG contents, also check the **True Scale (1:1)** property.* *This allows you to use the world coordinate system (mm) that was established during calibration, i.e. the coordinates in mm on your table or surface.* *If you don't check the **True Scale (1:1)** property, whatever content you provide will be displayed in the center of your **Image** visualization.* Provide your SVG content using standard SVG elements. Some typical examples you can use are <circle>, <ellipse>, <line>, <polyline>, <polygon>, <text> or <rect> (for a rectangle). For a full overview of SVG elements and their properties, see https://www.w3schools.com/graphics/svg_intro.asp. Encoding for SVG elements is UTF-8. ***Example*** *To render a yellow circle with radius 30 px and the middle of the circle located at coordinate (100,50), use* `<circle style='fill: yellow' cx='100' cy='50' r='30'/>`. **Important** While writing your SVG content, take into account the guidelines below: – Make sure the coordinates you provide in the SVG contents are within the coordinates of the **Image** visualization, otherwise you contents will not be visible. – Use simple quotes rather than double quotes. – Use the style attribute rather than the individual presentation attributes. See https://www.w3.org/TR/SVG/styling.html#StyleAttribute for more information. – For colors within the style attribute, you can either use rgb, rgba or hex color codes. Named colors are also supported. ***Note*** *If desired, you can also use a full SVG document, but in that case you need to take into account that the orientation of the Y-axis in SVG is different than the one used in the software. If you use SVG elements, the software converts the Y-axis for you, but for SVG documents you must take care of this conversion yourself.*
	The horizontal alignment of the image or video. This can be either left, center or right.
Maintain ratio	If checked, the image or video will always have the same width/height ratio, so it will not be stretched to fill up all available space.

Property	Description
True scale (1:1)	Allows to project images on a surface using scale 1:1, which is especially useful to project CAD drawings. Both vector (SVG and STL) and image formats (PNG, TIFF) are supported. Rendering on scale ignores the **Maintain ratio** and **Align** properties, as the coordinates in the source file are respected 1:1. The accuracy that can be achieved mostly depends on the amount of distortion introduced by the projector lens. The distortion depends on the type of projector, but typical values are +/- 0.5%. If the projector is not perpendicular to the surface, the margin increases with the angle. To achieve best results, accurate calibration of the projector is a must, and preferably manual projector calibration (from the ribbon) is used. ***Note*** *For more information on manual projector calibration, refer to* ***2.3 Other calibration options*** *on page 29.* As an example, a simple CAD drawing can be found in various formats in: *C:\Program Files\Arkite\Arkite Workstation\Help\CAD* 
Offset X (mm)	Specifies the offset of the image on the X axis. Only available if **True Scale (1:1)** is enabled.
Offset Y (mm)	Specifies the offset of the image on the Y axis. Only available if **True Scale (1:1)** is enabled.
Show in PDF	If checked, the image or a thumbnail of the video will be shown in the PDF export.

Button

A **button** visualization has the properties in the table below:

Property	Icon	Description
Set Variable		Check this checkbox if you want the state/value of a variable to change if the button is clicked.
Variable to Set		Variable that should be changed when clicking the button.
State to Set		Variable state that should be set when clicking the button. Enter the value you want to set for the selected variable(s). To insert dynamic content, click the icon at the location where you want to include the content and select the variable containing the dynamic content. Alternatively, you can also type the variable name between double braces, i.e. *{{Variable name}}*. You can also use a variable within a variable. For example, if the value of a variable **Index** is 01, then *{{Variable{{Index}}}}* will resolve into the value of the variable **Variable01**.

Property	Icon	Description
Set Active Variant		Check this checkbox if you want to set an active variant if the button is clicked. ***Note*** *For more information on working with variants, refer to section **5.6 Working with variants** on page 139.*
(variant to set)		Variant that should be set as the active one when clicking the button.
Process ControlDete		Check this checkbox if you want to use the button to perform an action on a particular process within your project.
Process / By Variable		Select one of the options below: – **Process:** select a specific process in the **Process** property – **By Variable:** use the value of the variable to define the process that needs to be manipulated
Process		Select the process you want to control with the button. Only available if the **Process** option is selected.
Variable for process name		The process with a name matching the value of this variable will be manipulated. Only available if the **By Variable** option is selected. ***Note*** *If the variable contains a value that does not match with a process name, the **Process Control** step will not be executed and an error will be added in the error tray.* *If the variable contains a value that matches with multiple process names, the process that was added first will be selected. It is a good practice to make sure your process names are unique.*
Action Type		Select the action you want to be performed on the selected process. You can choose to: – Reset a process – Pause a process – Pause and reset a process – Play a (paused) process – Restart a process, i.e. reset and play it – Go to the next step in a process – Go to the previous step in a process – Go to a specific step in a process.
Step/By Variable		Select one of the options below: – **Step:** select a specific step in the **Step** property – **By Variable:** use the value of the variable to define the step that needs to be activated Only available if the 'Go to specific step' option is selected in the **Action Type** property.
Step		Select the step you want to activate. Only available if the 'Go to specific step' option is selected in the **Action Type** property and if the **Step** option is selected.

Property	Icon	Description
Variable for step name		The first step in the process with a name matching the value of this variable will be activated. Only available if the ‘Go to specific step’ option is selected in the **Action Type** property and if the **By Variable** option is selected. ***Note*** *If the variable contains a value that does not match with a step name, no action will be executed and an error will be added in the error tray.* *If the variable contains a value that matches with multiple step names, the first step with a matching step name will be selected.*
Text Font Size	64	The font size of the displayed button label.
Text Color	A	The font color of the button label. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section **5.5.5 Managing visualization or DMX light colors** on page 130.*
Button text		Provide the text label you want show on the button, in any available language. To insert dynamic text, click the icon at the location where you want to include the text and select the variable containing the dynamic text. Alternatively, you can also type the variable name between double braces, i.e. *{{Variable name}}.* You can also use a variable within a variable. For example, if the value of a variable **Index** is 01, then *{{Variable{{Index}}}}* will resolve into the value of the variable **Variable01**. ***Tip*** *For long texts, click the icon to enlarge the text box.* ***Note*** *To add or remove languages, refer to section **15.2 Managing languages** on page 237.*
Image		Image to be displayed on the button
Maintain ratio		If checked, the image will always have the same width/height ratio, so it will not be stretched to fill up all available space.
Enabled		Select one of the options below from the **Enabled** dropdown list: – **On**: the screen button will always be enabled. The operator will be able to click or press it, and the action(s) defined in the **Action** properties will be executed. – **Off**: the screen button will always be disabled. The button will have a reduced opacity. No actions will be executed when the operator tries to click/press it. – **Based on Variable**: the enabled/disabled state of the screen button will be based on the current state of the variable provided in the **Variable for enabled** property.
Variable for enabled		Only available if the **Based on Variable** option is selected in the **Enabled** dropdown list. The state of this variable will determine whether the screen button will be enabled: – If the state of the variable is ON, ENABLED or 1, the screen button will be enabled. – If the state of the variable is OFF, DISABLED or 0, the screen button will be disabled.

Property	Icon	Description
Button Fill Color		The fill color of the button. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section* ***5.5.5 Managing visualization or DMX light colors*** *on page 130.*
Button Border Color		The border color of the button. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section* ***5.5.5 Managing visualization or DMX light colors*** *on page 130.*

Counter

A **counter** visualization has the properties in the table below:

Property	Icon	Description
Counter actuator		Select the counter for which you want to display the value.
Display Spark rate instead of value		If checked, the frequency by which the counter value is changed is displayed instead of the value itself. ***Example*** *Suppose you have a repeating process consisting of a* ***Counter*** *step and a* ***Delay*** *step configured at 2000 ms. In that case, if you choose to display the value of the* ***Counter****, the value displayed in your visualization will go up every 2 seconds. If you choose to display the spark rate, the value displayed in your visualization will be 30 per minute.*
Text Font Size		The font size of the displayed counter value.
Text Color		The font color of the displayed counter value. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section* ***5.5.5 Managing visualization or DMX light colors*** *on page 130.*

Progress bar

A **progress bar** visualization has the properties in the table below:

Property	Icon	Description
Based on process?		If checked, the progress bar visualization will be based on a process. The current value is the active step, the max value of the progress bar is the total number of steps in the process. If unchecked, the progress bar visualization will be based on the state of the variables defined in the **Variable for current value** and **Variable for max value** properties.
Process		Process the progress bar visualization is defined on.
Variable for current value		Specifies the variable for which the state will be used as current value. The state of the variable should be numeric.

Property	Icon	Description
Variable for max value		Specifies the variable for which the state will be used as the max value (100%) of the progress bar. The state of the variable should be numeric.
Orientation		Determines the orientation of the progress bar: left to right (HORIZONTAL) or bottom to top (VERTICAL).
Mirror		When checked the progress bar will be mirrored: right to left in case of HORIZONTAL orientation, top to bottom in case of VERTICAL orientation.
Progress Color		The color of the progress indicator. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section **5.5.5 Managing visualization or DMX light colors** on page 130.*

Operator Input

A **Operator Input** visualization has the properties in the table below:

Property	Icon	Description
Variable for text input		Variable in which the operator's answer is stored. As soon as the operator starts typing in the field, the variable is updated.
Show current value?		If checked, the current value of the variable defined in the **Variable for text input** property is shown in the input field.
Text Font Size		The font size of the displayed text.
Text Color		The font color of the text. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section **5.5.5 Managing visualization or DMX light colors** on page 130.*

Note

*If no physical keyboard is available, also add a **Keyboard** visualization to allow the operator to enter text in the **Operator Input** visualization.*

Keyboard

The **Keyboard** visualization has the properties in the table below:

Property	Description
Keyboard type	Select type of keyboard that will be displayed: – **Alphanumeric**: characters and numbers – **Alphabetic:** characters only – **Numeric:** numbers and arithmetic operators only

Sensor stream with detections (Validate and Vision license only)

A **sensor stream** visualization has the properties in the table below:

Property	Icon	Description
Mirror horizontally		If checked, the sensor stream will be mirrored horizontally to compensate the mirror caused by the depth sensor.
Mirror vertically		If checked, the sensor stream will be mirrored vertically to compensate the mirror caused by the depth sensor.
Detections		Detections that should be displayed in the sensor stream visualization.
Color for 'OFF'		The color for an inactive detection. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section **5.5.5 Managing visualization or DMX light colors** on page 130.*
Color for 'ON'		The color for an active detection. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section **5.5.5 Managing visualization or DMX light colors** on page 130.*

5.5.3 Adding visualizations

To add a new visualization, execute the steps below:

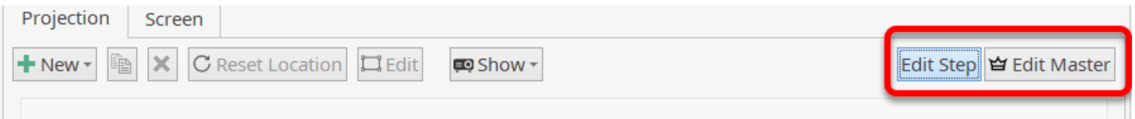
1. In the **Processes** tab, select the process and the process step for which you want to add a visualization.

   Note

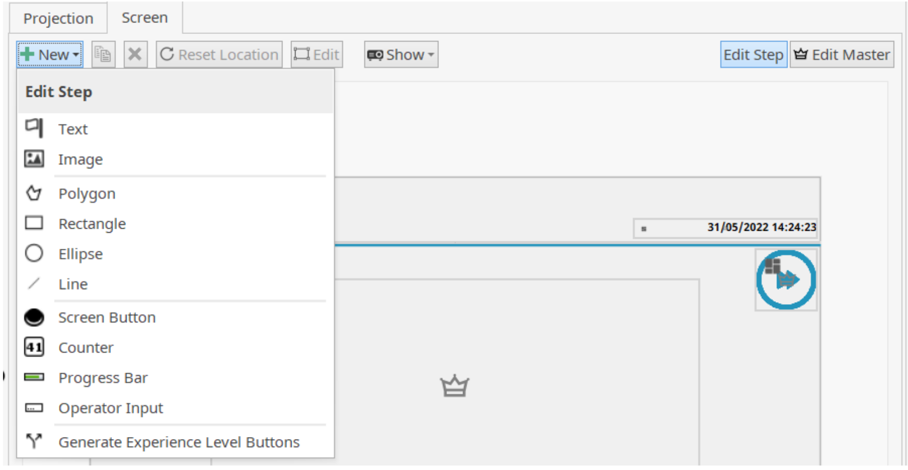
   If you want to add a visualization that is visible regardless of the process or step, it does not matter which process and step you select.

2. Click the display subtab on which you want to add a visualization, **Projector** or **Screen**.
3. To add a visualization that is only shown when the selected step is active, click **Edit Step**.

   To add a visualization that is always shown, regardless of the process or process step, click **Edit Master**.



4. Click the **New** button and select the visualization of your choice.



Tip

*You can also select an existing visualization and click **Duplicate**.*

5. Define the size and position of the visualization using your mouse.

Tip

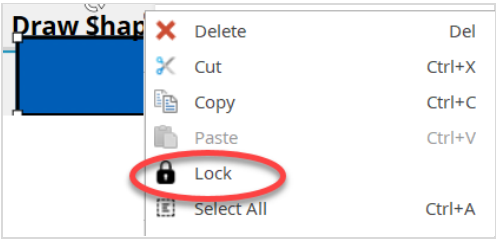
While dragging the visualization with your mouse, visual guidelines will appear that will allow you to easily align the visualization. The guidelines below will appear:

- *Guidelines towards the **middle of the screen** will appear when you reach the middle of the screen with an edge or with the middle of the visualization, both horizontally and vertically.*
- ***Object to object alignment lines** appear to help you to align two visualizations with each other, both horizontally and vertically. This way, you can align the top, the bottom or the middle of the visualizations with each other.*
- *Arrows appear to help you to position two visualizations at the **same distance from the edge or the center** of the screen.*
- *When you have more than two visualizations, arrows appear when the distance between visualizations 1 and 2 is the same as between visualizations 2 and 3. This helps you to ensure there is the same distance between visualizations.*
- *When you are changing the size of an object, lines appear inside the objects when they have the same length or width. This helps you to make the objects have the **same size**.*

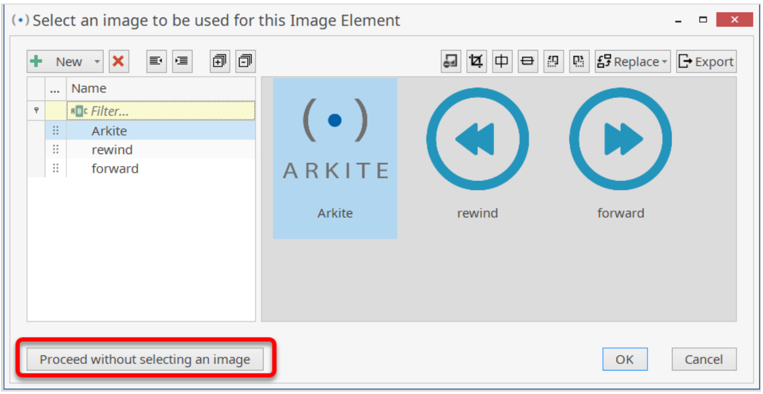
6. In the **Properties** pane, update the properties of the visualization if needed.

Note

- *The properties that are available depend on the type of visualization you have added. For an overview of visualization types and their properties, refer to sections **5.5.1 Visualization types** on page 116 and **5.5.2 Visualization properties** on page 117.*
- *To protect a visualization from being moved or edited by accident, right-click it and select **Lock**.*



- *When you want to add an **Image** visualization for which you want to use a variable, click **New – Image** and then click **Proceed without selecting an image**. In the properties of the **Image** visualization, you can define the variable to use. For more information, refer to section **5.5.2 Visualization properties** on page 117 .*



5.5.4 Editing visualizations

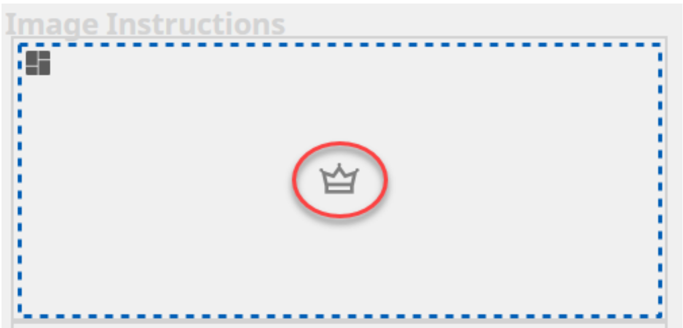
To edit an existing visualization, execute the steps below:

1. In the **Processes** tab, select the process and the process step for which you want to edit a visualization.

   Note

   *To edit a visualization that is visible in all processes and all steps, click **Edit Master** first.*

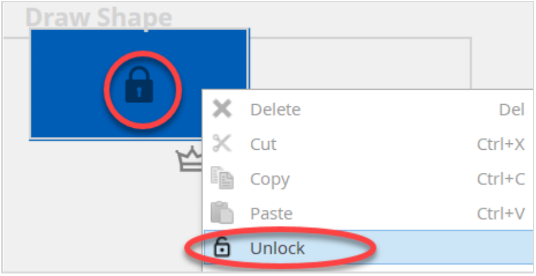
   Visualizations on the master are marked with a crown icon:

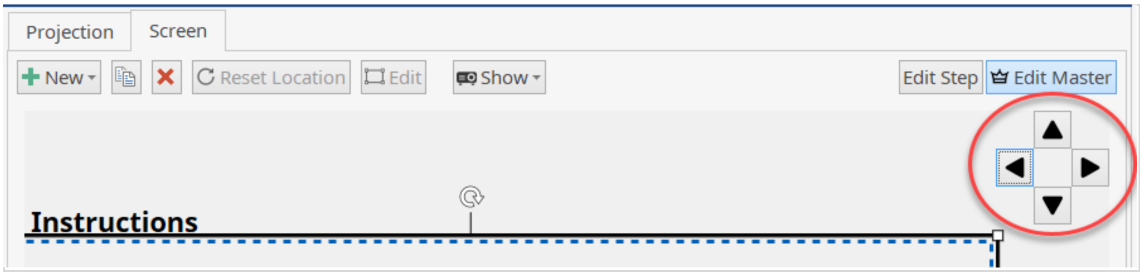
2. Click the display subtab on which you want to edit a visualization, **Projector** or **Screen**.
3. Select the visualization you want to edit.

   Note

   - *You cannot edit a visualization that is locked. First right-click the visualization and select **Unlock**.*



- *You can select multiple visualizations by holding the **Ctrl** button and clicking the visualizations of your choice with your mouse, or by holding the left mouse button and dragging a rectangle over the visualizations of your choice. Press **Ctrl-A** to select all visualizations.*
- *You can zoom in and out using your mouse wheel. When zoomed in, arrows appear to navigate around in your display.*



4. For draw shapes:
   - Change the size and position of the visualization by dragging the shape or the angular points with your mouse.
   - To change the shape of the visualization, click the **Edit** button above the image and move the angular points.
5. For all shapes, update the properties of the visualization in the **Properties** pane if needed.

   Note

   *The properties that are available depend on the type of visualization you have added. For an overview of visualization types and their properties, refer to sections **5.5.1 Visualization types** on page 116 and **5.5.2 Visualization properties** on page 117.*

To delete an existing visualization, select it and click **Delete**.

5.5.5 Managing visualization or DMX light colors

Several color properties can be set for a visualization or for a DMX light, in case a DMX device is connected through a DMX communication module.

Every color property can be defined in four separate ways:

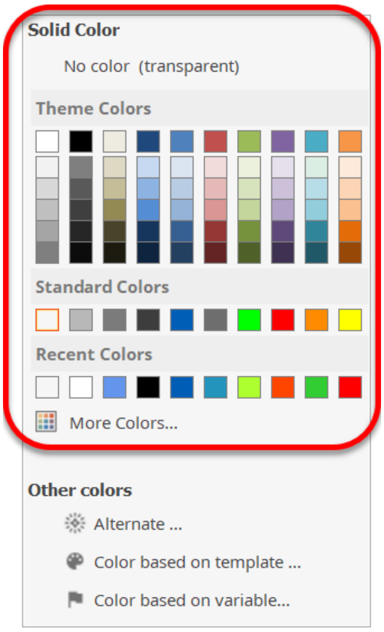
- Setting a fixed solid color
- Selecting two alternating colors
- Defining the color based on the value of a variable
- Defining the color based on a template

Note

- *If you define a color based on the value of a variable, you can manage the color separately from the colors for all other visualizations.*
- *If you define a color based on a template, all visualizations linked to that template will automatically change colors when the template is updated.*

Setting a fixed solid color

If you set a fixed, solid color, the visualization will always have the same color. To do so, click the color icon and select a color from the menu.



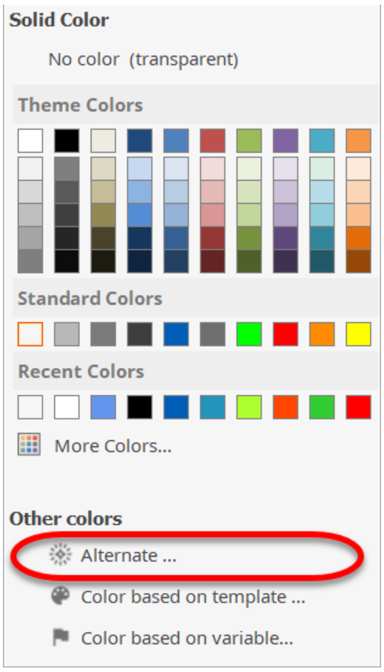
Note

*Click **More Colors...** to define a color based on the RGB or HSB model.*

Selecting two alternating colors

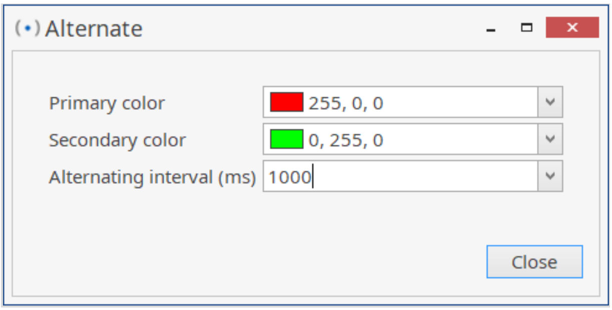
To make the visualization alternate between two colors, execute the following steps:

1. Click the color icon.
2. Under **Other colors**, select **Alternate...**



3. In the **Alternate** window:
   - Select your first color in the **Primary color** dropdown
   - Select your second color in the **Secondary color** dropdown
   - Select an **Alternating interval** in milliseconds (250, 500 or 1000)

   In the example below, the visualization's color will alternate between red and green every second:

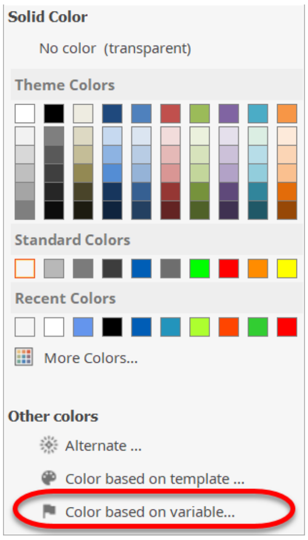


4. Click **Close**.

Defining the color based on the value of a variable

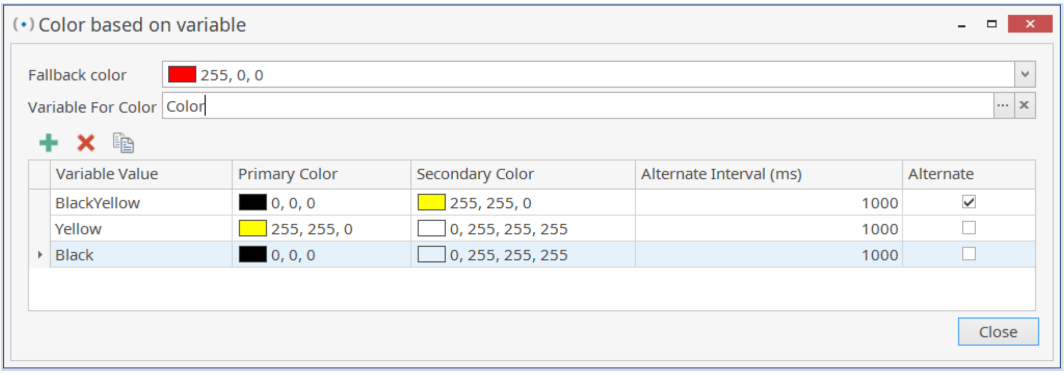
To define the color based on the value of a variable, execute the following steps:

1. Click the color icon.
2. Under **Other colors**, select **Color based on variable...**



3. In the **Color based on variable** window:
   - In the **Fallback color** property, indicate which color the visualization should have if the variable has a value that is not defined.
   - In the **Variable for color** property, specify the variable that contains the values to define the color of the visualization.
   - Define one or more values and the corresponding color(s).

In the example below, the visualization's color will be yellow when the value of the **Color** variable is *Yellow* and black when the value of the **Color** variable is *Black.* When the value of the **Color** variable is *BlackYellow*, the color will alternate between black and yellow every second. When the **Color** variable has another value, the visualization's color will be red (i.e. the fallback color).



Note

- *In case the **Alternate** checkbox is checked, the color of the projection will alternate between the **Primary Color** and the **Secondary Color**.*
- *In case the **Alternate** checkbox is not checked, the **Primary Color** will always be used.*

4. Click **Close**.

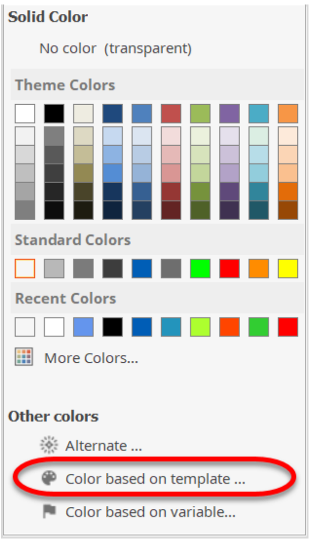
   Note

   *When there is a color defined for the variable value ON, this color will be used in the UI and when your project is in **Edit** mode. In **Run** mode, the color as specified by the variable will be used.*

Defining the color based on a template

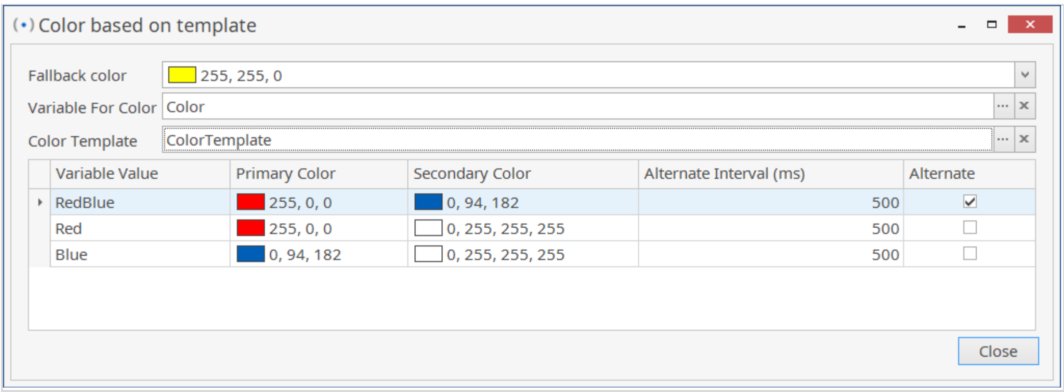
To define the color based on the value of a variable, execute the following steps:

1. Click the color icon.
2. Under **Other colors**, select **Color based on template...**



3. In the **Color based on template** window:
   - In the **Fallback color** property, indicate which color the visualization should have if the template does not define a color.
   - In the **Variable for color** property, specify the variable that contains the values to define the color of the visualization.
   - In the **Color Template** property, select the template to use.

   In the example below, the visualization's color will be red when the value of the **Color** variable is *Red* and blue when the value of the **Color** variable is *Blue.* When the value of the **Color** variable is RedBlue, the color will alternate between red and blue every 500 milliseconds. When the **Color** variable has another value, the visualization's color will be yellow (i.e. the fallback color).



Note

- *In case the **Alternate** checkbox is checked, the color of the projection will alternate between the **Primary Color** and the **Secondary Color**.*
- *In case the **Alternate** checkbox is not checked, the **Primary Color** will always be used.*

4. Click **Close**.

Note

- *When there is a color defined for the variable value ON, this color will be used in the UI and when your project is in **Edit** mode. In **Run** mode, the color as specified by the template will be used.*
- *When you define a color based on a template, you cannot change the values and their corresponding colors in that template. For instructions on how to manage color templates, refer to section **14.3 Managing color templates** on page 233. Be aware that changing a color template changes the color of all visualizations linked to that color template.*

5.5.6 Managing text and image instructions

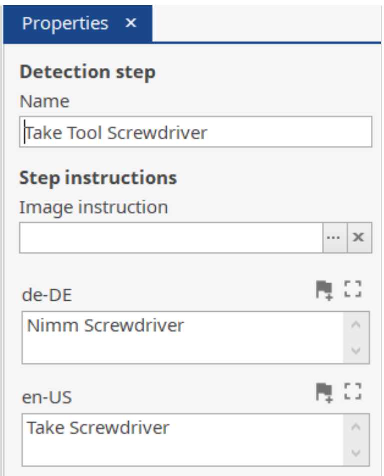
When creating a process, the Arkite software will generate visualizations to display text and image instructions. At the same time, two variables are created for text and image instructions. These variables will always contain the instructions for the active step and make sure the right instructions are displayed at the right time.

To define the instructions that will be shown for a particular step, execute the steps below:

1. Navigate to the **Processes** tab.
2. Under **Steps**, select the step for which you want to add instructions.
3. In the **Properties** pane, under **Step instructions**, select the image or video you want to be displayed.
4. Provide a text instruction you want to be displayed in all available user languages.

   For detection steps, default instructions are proposed.

   To insert dynamic text, click the icon at the location where you want to include the text and select the variable containing the dynamic text. Alternatively, you can also type the variable name between double braces, i.e. {{Variable name}}. You can also use a variable within a variable. For example, if the value of a variable **Index** is 01, then *{{Variable{{Index}}}}* will resolve into the value of the variable **Variable01**. For long texts, click the icon to enlarge the text box.

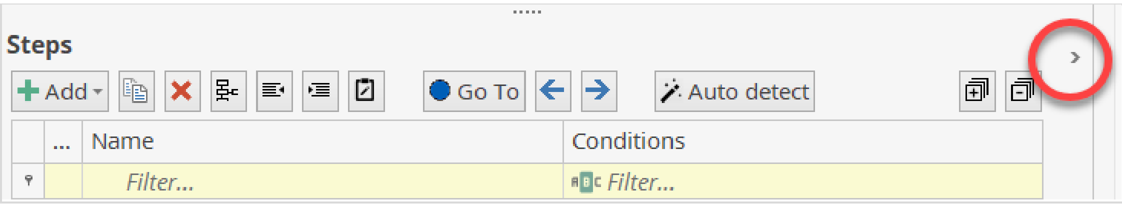


Note

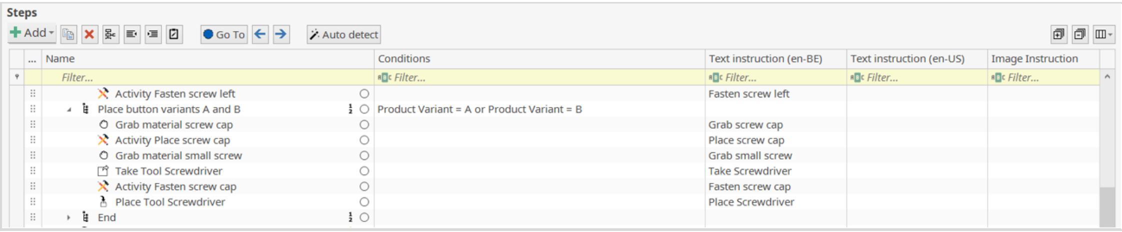
*To add or remove languages, refer to section **15.2 Managing languages** on page 237.*

Tip

*To have an overview of all instructions for all steps in a process, you can also click the right arrow on top of the **Steps** overview.*



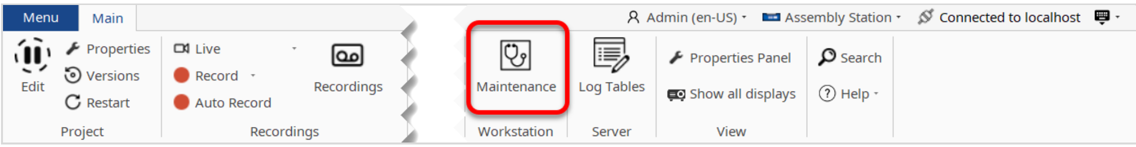
Additional columns will appear in which you can see and edit text and image instructions for every step.



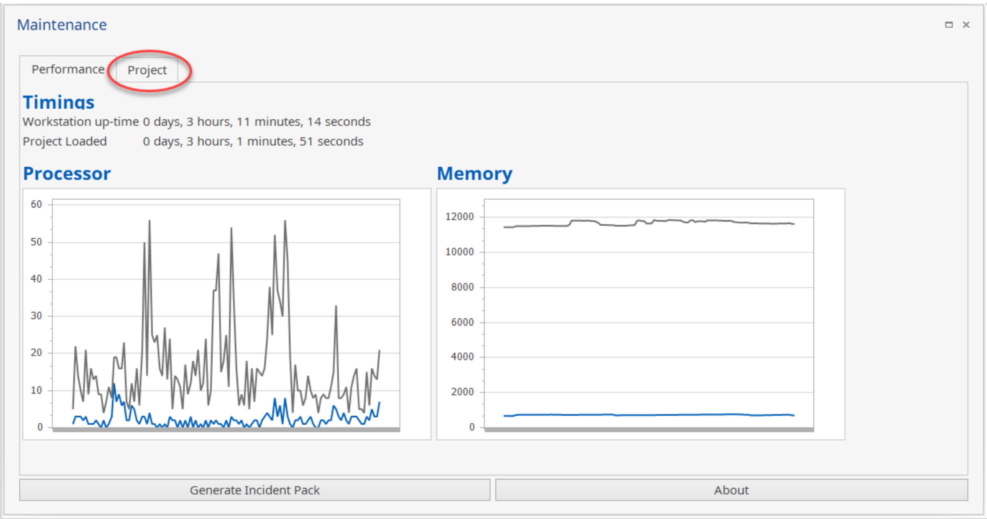
5.5.7 Managing projection settings

During the execution of your assembly process, the projections linked to the detections used in your process steps will be activated one after another. To configure the exact behavior of your projections, execute the steps below:

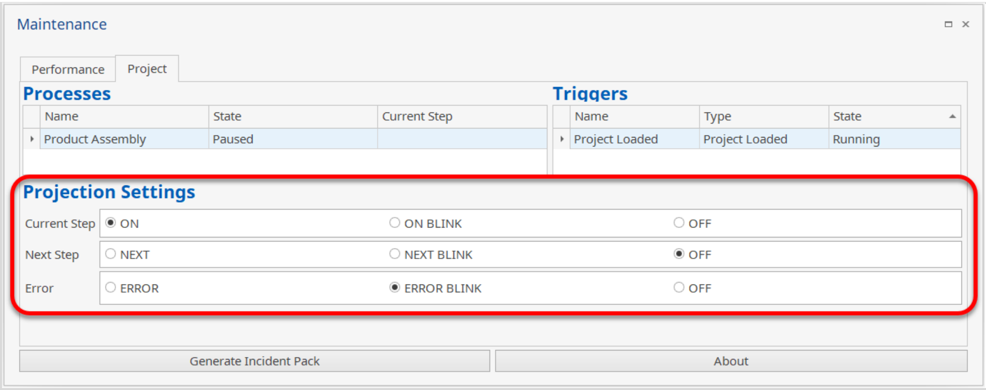
1. In the ribbon, in the **Workstation** group, click the **Maintenance** button.



2. Click the **Project** tab.



3. Under **Projection Settings**, change the settings for the current step, the next step and errors.

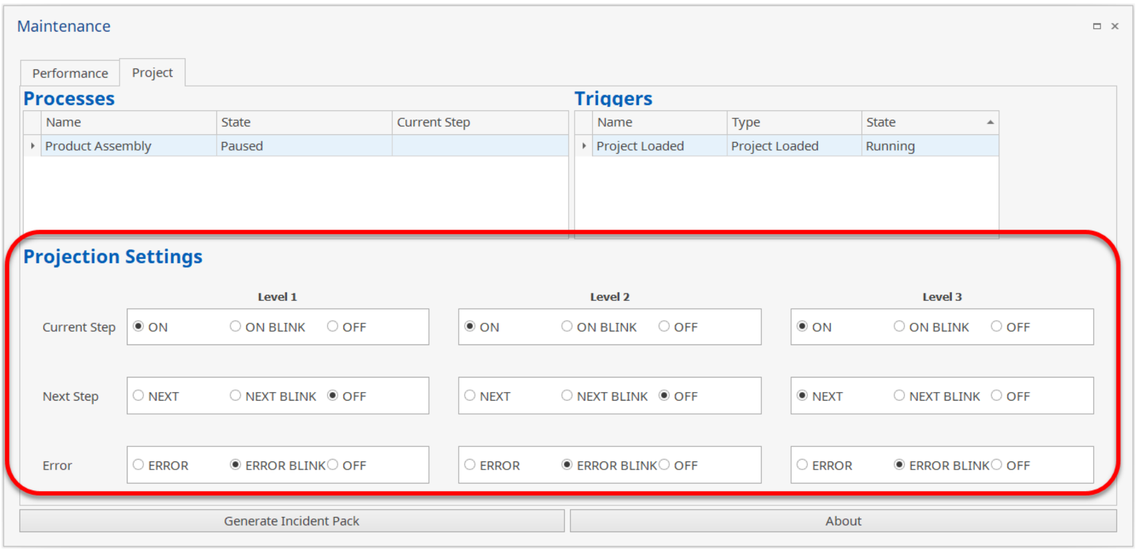


The options are:

- For the current step in your process:
  ON: a projection will be shown
  ON BLINK: a projection will be shown and it will blink
  OFF: no projection will be shown
- For the next step in your process:
  NEXT: a projection will be shown
  NEXT BLINK: a projection will be shown and it will blink
  OFF: no projection will be shown
- For errors during your process:
  ERROR: a projection will be shown
  ERROR BLINK: a projection will be shown and it will blink
  OFF: no projection will be shown

Note 1

*In case the **Experience Level** functionality is enabled, the settings described above can be set independently for each experience level.*



*For more information on the **Experience Level** functionality, refer to chapter **7 Experience levels** on page 194.*

Note 2

*To change the colors of these projections, click the **Color Templates** button in the ribbon and select the **Detection colors** template. There you can update the colors for the projection settings mentioned above. For more information, refer to section **14.2 The Detection Colors template** on page 232.*

5.5.8 Adding experience level buttons

When the **Experience Level** functionality is enabled and **Manual** experience levels are selected, you can add experience level buttons to a **Screen** display to allow an operator to select his experience level.

Note

*For more information on the **Experience Level** functionality, refer to chapter **7 Experience levels** on page 194.*

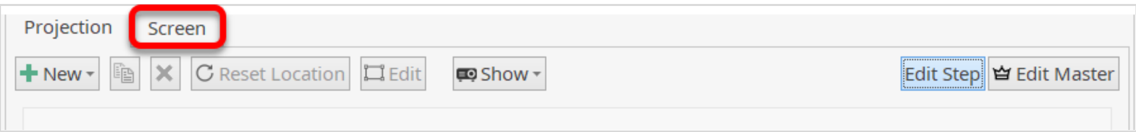
To add experience level buttons, execute the steps below:

1. In the **Processes** tab, select any process.

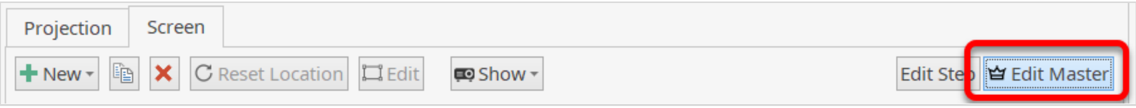
   Note

   Typically, you'll want to allow an operator to be able to select or change his experience level at any time in the process. However, if you want the experience level buttons only to be displayed when a specific process step is active, first select the process and process step of your choice.

2. Click the **Screen** display subtab.



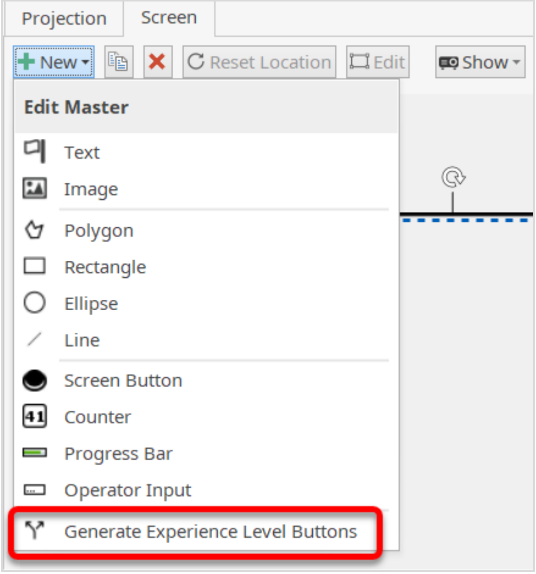
3. Click **Edit Master**.



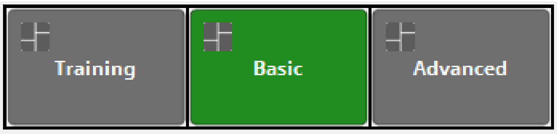
Note

*If you want the experience level buttons only to be displayed when the selected process step is active, click **Edit Step**.*

4. Click the **New** button and select **Generate Experience Level Buttons**.



For each available experience level, a button is added to the screen. The button for the current experience level is displayed in green, the other buttons are displayed in grey.



Clicking a button changes the experience level.

Warning

*When the **Experience Level** functionality is disabled, you have to manually remove the experience level buttons again.*

5.6 Working with variants

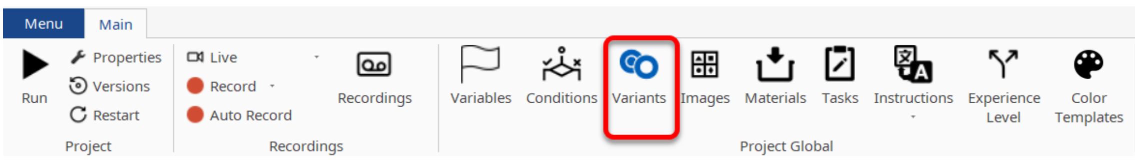
What?

Variants refer to different flavors of the same product. Product variants have a lot of common attributes, but some attributes will be specific. Often the assembly process will also contain a lot of common steps. Variants allow you to efficiently cater for different variants within the same process. It allows you to indicate which steps are common and which steps should only be executed for specific variant(s).

Adding variants

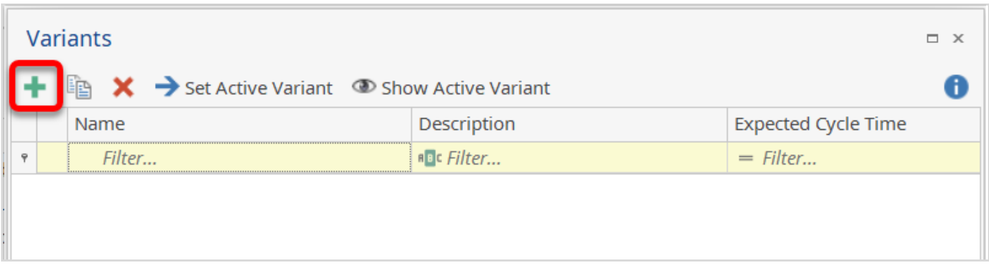
To add a variant, execute the steps below:

1. In the ribbon, in the **Project Global** group, click the **Variants** button.



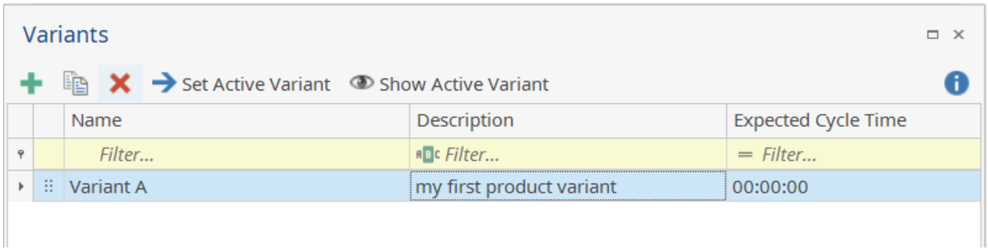
The **Variants** window appears.

2. To add a variant, click the button.



A variant is added to the list.

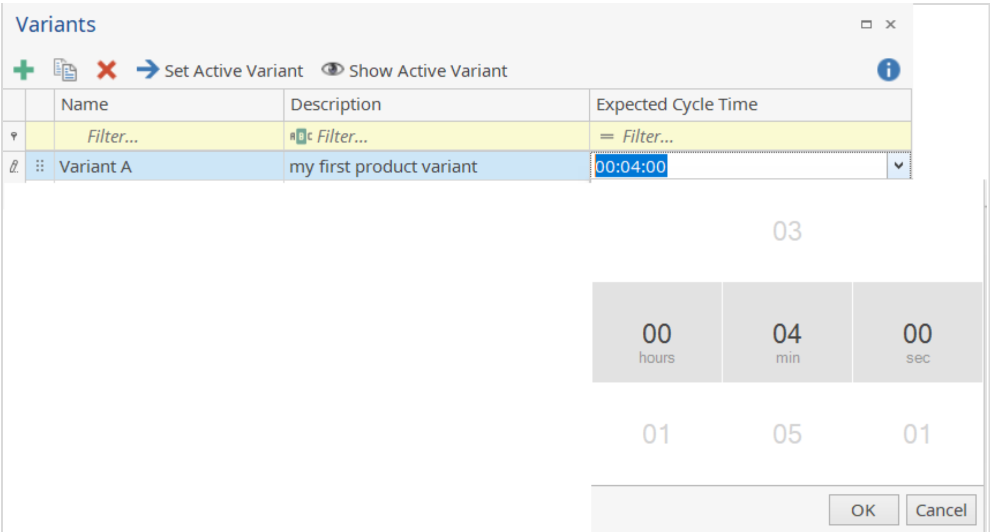
3. Provide a name and a description for your variant.



Note

Keep the name for your variant as short as possible, since it will be used as a column header in the process steps section.

4. Optionally, provide an **Expected Cycle Time** for the variant. This is the estimated time it takes to assemble one product. This parameter will be used in the **Cycle Times** dashboard.



Note

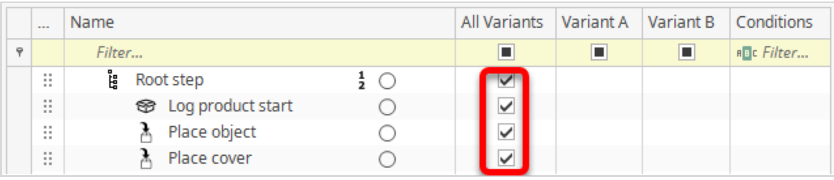
For more information about dashboarding, refer to chapter ***11 Dashboarding*** *on page 215.*

Linking variants to process steps

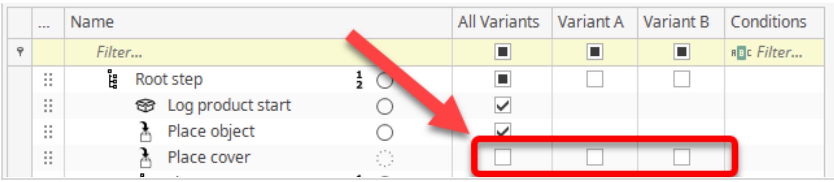
After you have added a variant, you can link it to a step in your process. To do so, execute the steps below:

1. Navigate to the **Processes** tab.
2. In the **Processes** section, select the process that contains one or more steps you want to link to a variant.
3. In the **Steps** section, select the **Variants and Conditions** radio button.

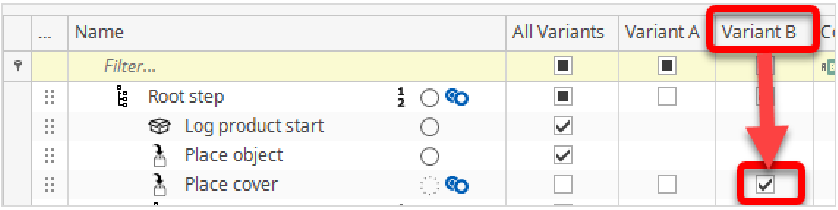
4. Go to the step you want to link to a variant.
   By default, each step is linked to all variants.



5. To link a step, for example the **Place cover** step in the example below, to specific variant(s) only:
   - Uncheck the checkbox in the **All Variants** column.
     Checkboxes in the columns for each variant become visible



   - Check the checkbox(es) in the column(s) for the variant(s) you want to link the step to.
     For example, check the checkbox for Variant B for the **Place cover** step.



Note

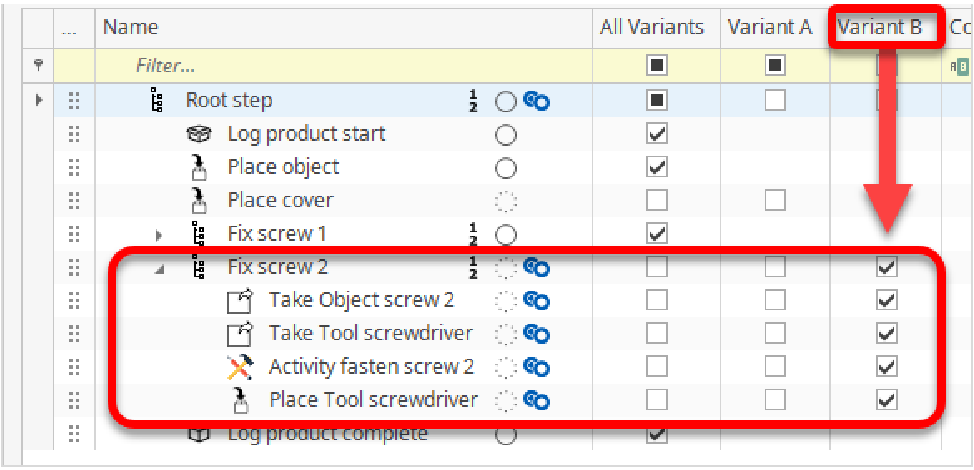
When you link a variant to a composite step, it is automatically linked to all of its child steps. Afterwards, you can unlink any individual child step as required. Composite steps for which some child steps are linked to a variant, and others are not, have a checkbox in the state for that variant.

Example

In the example below, we have a small process in which a cover is fastened to an object by means of one or two screws.

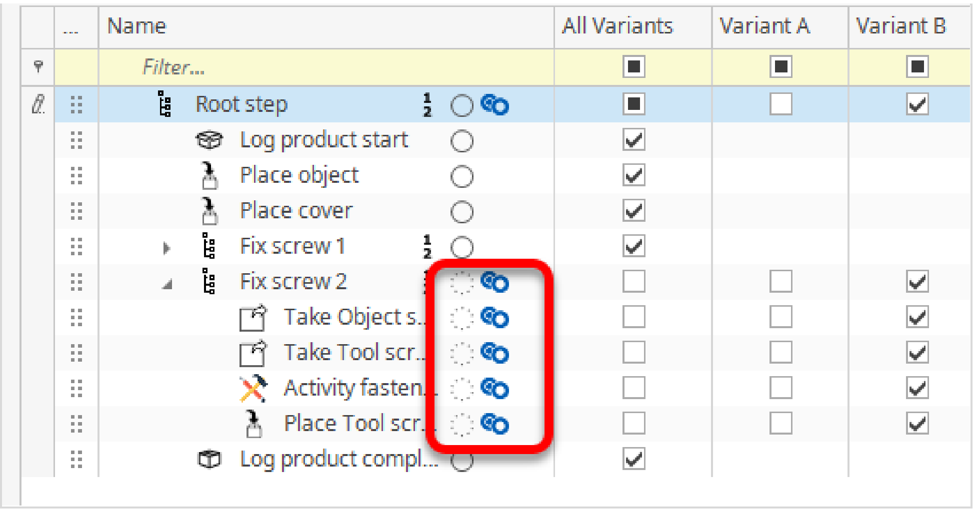
For Variant A, only 1 screw is used. For Variant B, 2 screws are used.

The process is the same for both variants, except for the steps related to the seconds screw. These are only linked to Variant B.



When the **Variants** columns are not shown, you can still see which steps are linked to specific variant(s) and therefore will not always be executed. They are marked by a icon.

You can also see which steps that are linked to a variant will be executed in the current situation (icon), and which will be skipped (icon). In the example below, Variant A is currently active which means the steps linked to variant B will be skipped.



Managing variants

To manage variants, click the **Variants** button in the **Project Global** group in the ribbon.

The **Variants** window appears.

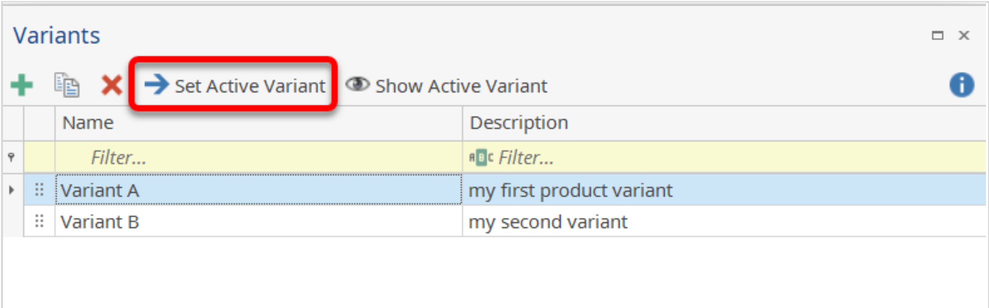
Select a variant and click to duplicate it.

Select a variant and click to delete it.

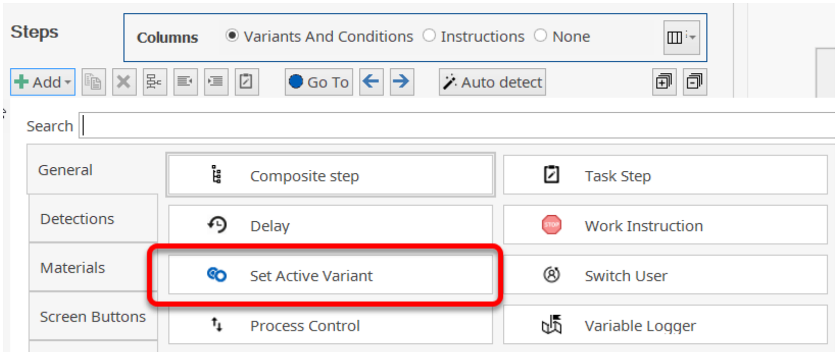
Setting a variant

There are several ways to mark a variant as the active one:

- Set the active variant as a project administrator, while managing your project:
  - Click the **Variants** button in the **Project Global** group in the ribbon.
  - Select the variant of your choice and click the **Set Active Variant** button.



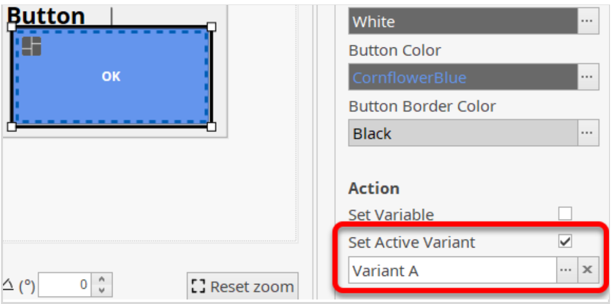
- Let the system set the active variant as part of the assembly process:
  Add a process step of the type **Set Active Variant.**



Note

*For more information, refer to section **5.4.2 Adding process steps** on page 111.*

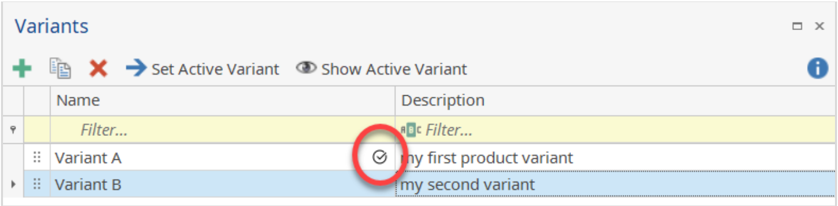
- Let the operator set the active variant himself:
  Add a **Button** visualization to a **Projector** or **Screen** display and check the **Set Active Variant** property.



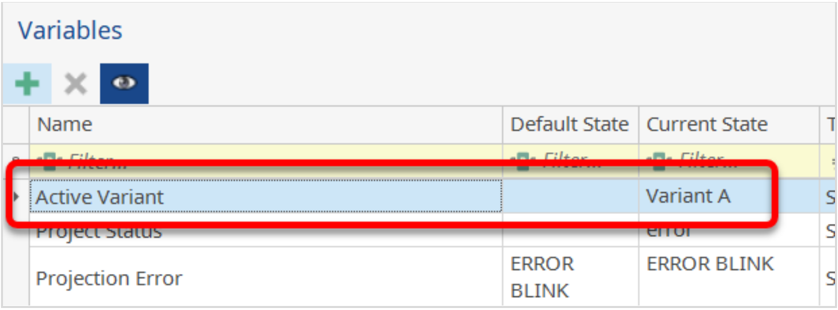
Note

*For more information, refer to section **5.5.3 Adding visualizations** on page 127.*

In the **Variants** window, the active variant is indicated with a ⊘ icon.



The active variant is stored in the **Active Variant** variable. In the **Variants** window, click **Show Active Variant** to open the **Variables** window with the **Active Variant** variable selected.



5.7 Working with conditions

What?

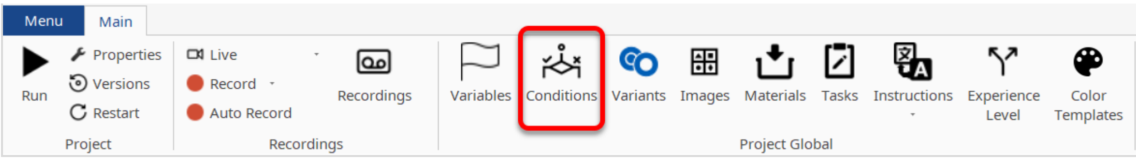
Conditions can be used to specify whether a process step should be executed or not. Any individual or composite process step can be linked to one or more conditions. Only if at least one of the conditions holds, the step will be executed.

A condition can consist of one or more subconditions.

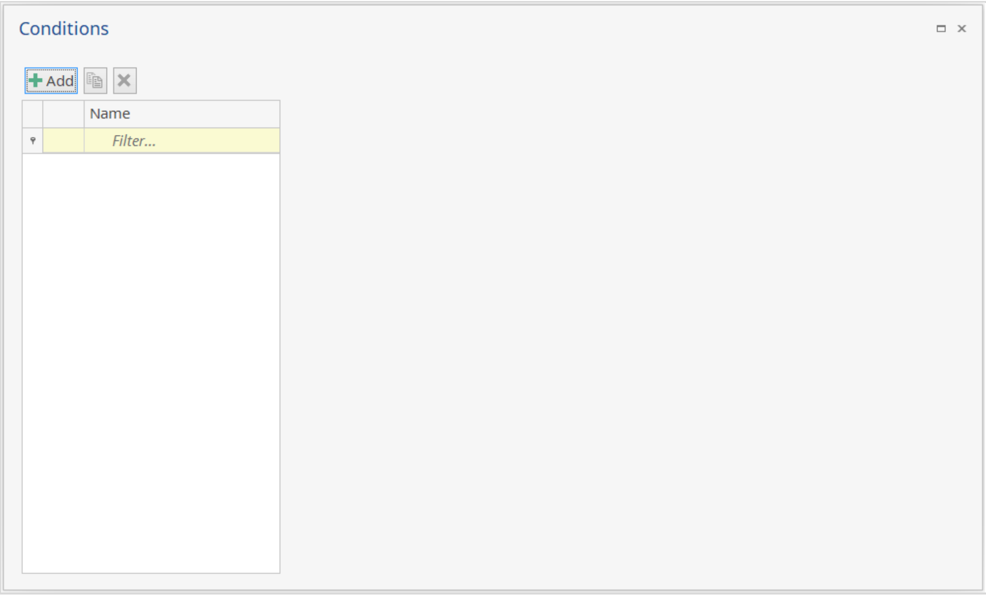
Adding conditions

To add a condition, execute the steps below:

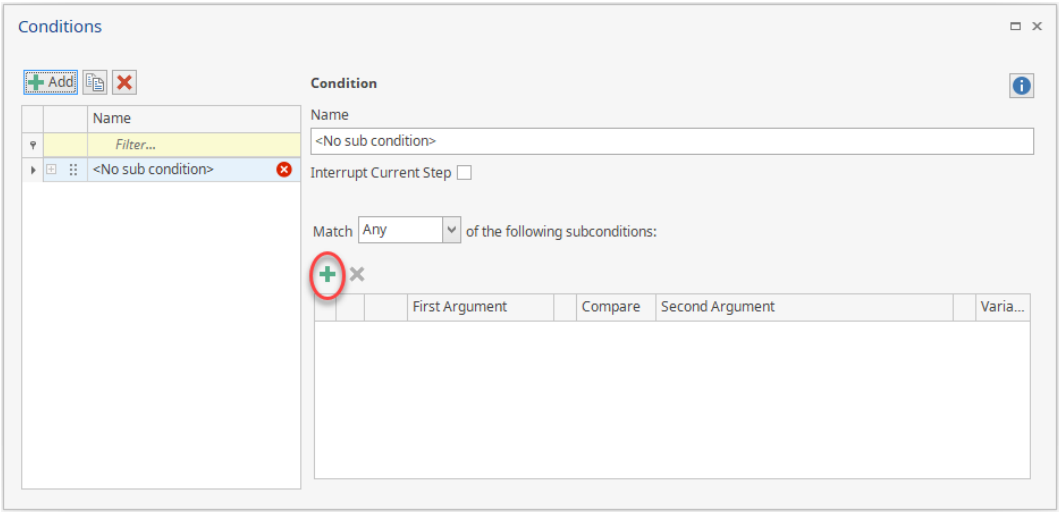
1. In the ribbon, in the **Project Global** group, click the **Conditions** button.



The **Conditions** window appears.



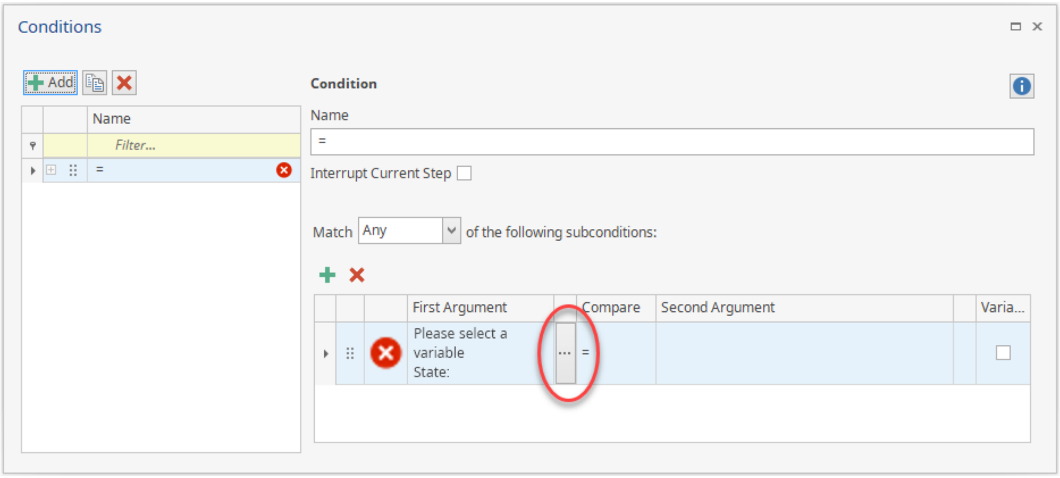
2. To add a condition, click the **Add** button.
   A condition is added to the list.
3. To add a subcondition, click the button.



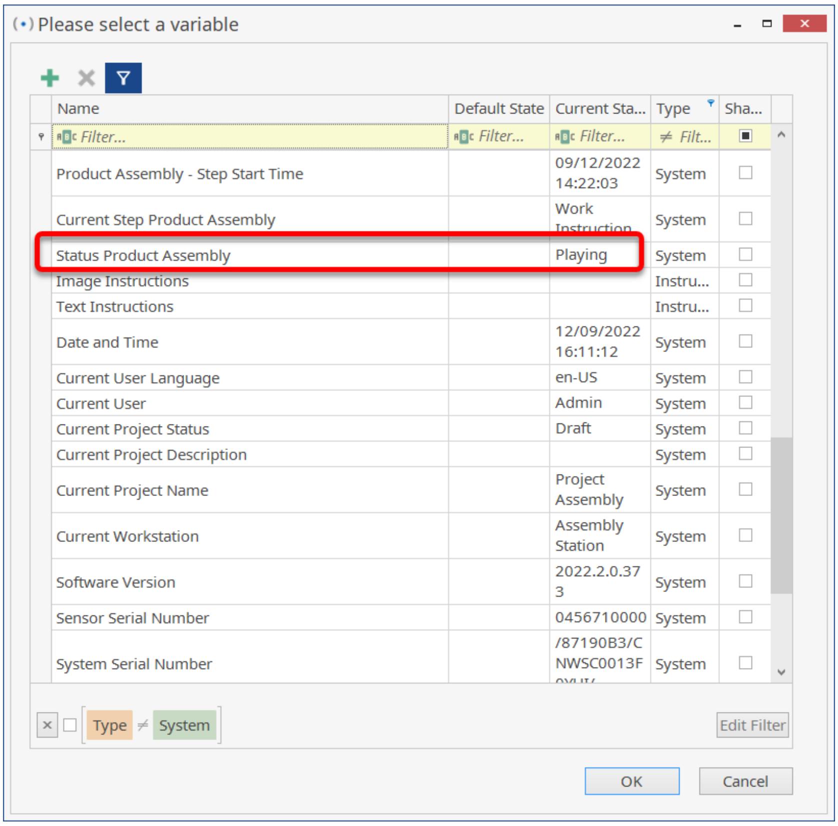
A subcondition is added to the list, consisting of a

- **First argument**: this should always be a variable.
- **Compare method**
- **Second argument**: this can be a variable or a string.

4. To define the first variable, click the **...** button.



5. Select a variable and click **OK**.
   In the example below, we use the variable **Status Product Assembly**.



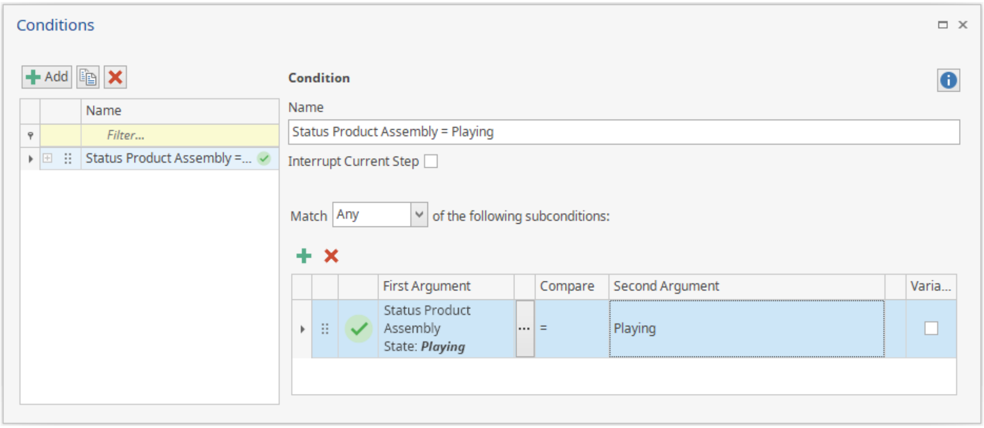
Note

*For more information about working with variables, refer to chapter **8 Variables** on page 199.*

6. Select a compare method.

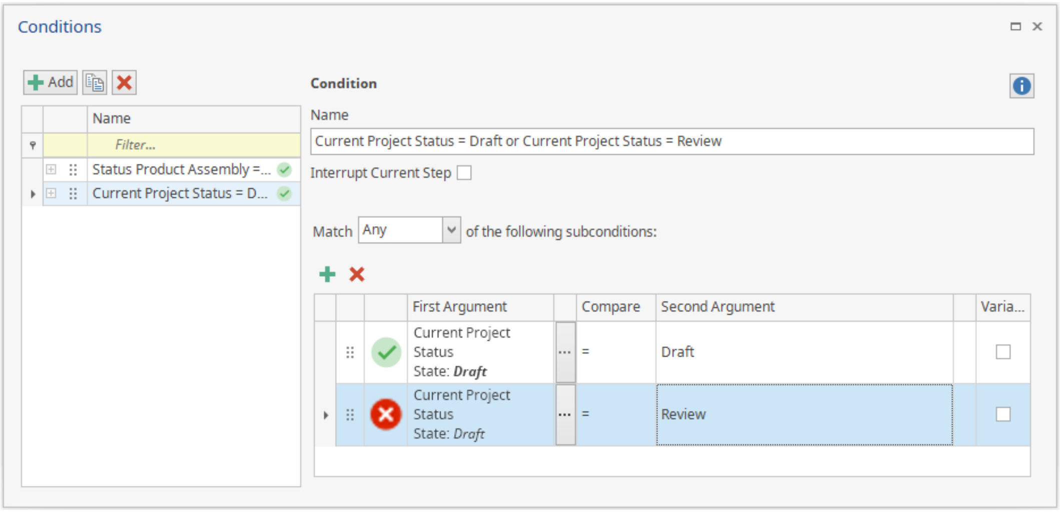
7. Define the second argument:
   - To compare the state of the variable in the **First Argument** with a fixed string, type it in the **Second Argument** column.
   - To compare the state of the variable in the **First Argument** with the state of another variable, first check the checkbox in the **Variable** column and then in the **Second Argument** column, click the **...** button and select a variable.

     In the example below, we have a created a subcondition where the state of the variable **Status Product Assembly** needs to be **Playing** for the condition to be true.



8. Optionally, you can create additional subconditions. You can define whether all, none or any (=at least one) of the subconditions need to be true for the condition to be true.

   In the example below, we have a created a condition where the state of the variable **Current Product Status** needs to be **Draft** (first subcondition) or **Review** (second subcondition) for the condition to be true.



Note

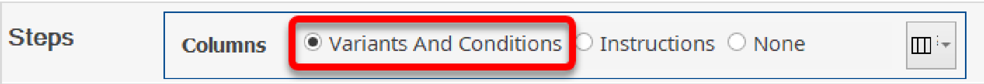
*The **Name** of a condition is automatically defined by the subconditions you create. However, you can edit it.*

*By checking the **Interrupt Current Step** checkbox, you can indicate a process step needs to be interrupted when the condition becomes invalid while the step is being executed. For example, a receive communication step will stop waiting for incoming data when its condition no longer holds.*

Linking a condition to a process step

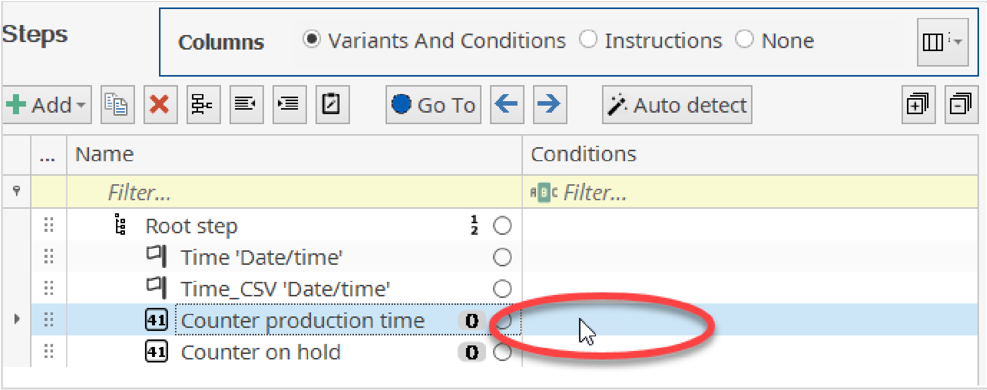
After you have created a condition, you can link it to a step in your process. To do so, execute the steps below:

1. Navigate to the **Processes** tab.
2. In the **Processes** section, select the process that contains one or more steps you want to link to a condition.
3. In the **Steps** section, select the **Variants and Conditions** radio button.

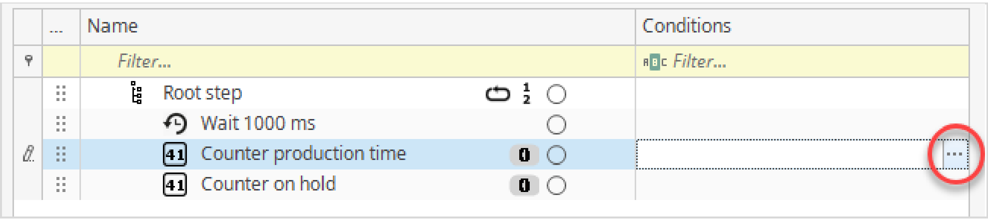


4. Go to the step you want to link to a condition and click in the **Conditions** column.

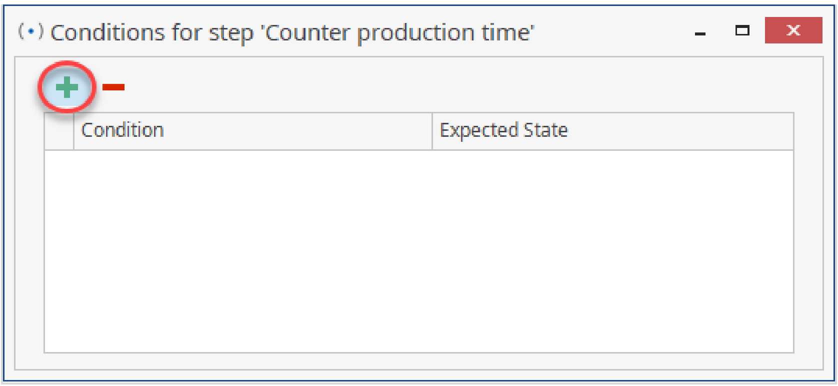
   For example, to link the step Current production time to a condition, click on the location indicated in the screenshot below.



5. Click the **...** button at the end of the column.

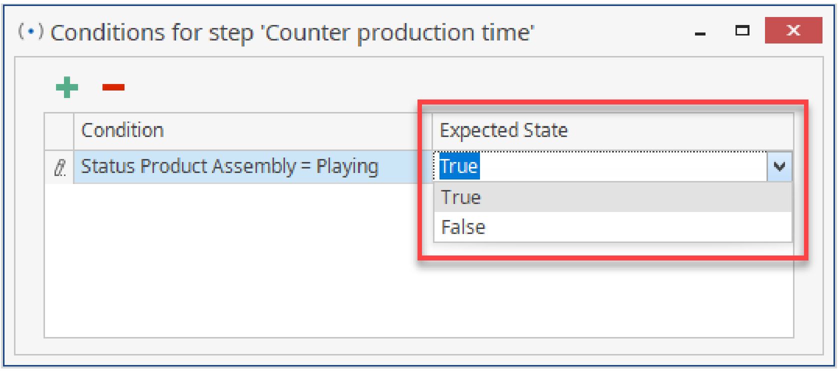


6. In the popup window, click the  button.



7. Select the condition you want to add to the step and click **OK**.

8. In the **Expected State** column, select if you want the step to be executed when the condition is **True** or when it is **False**.



9. Optionally, you can add more conditions to your process step. A step will be executed if at least one of the linked conditions hold.
10. When you have finished, close the popup window.

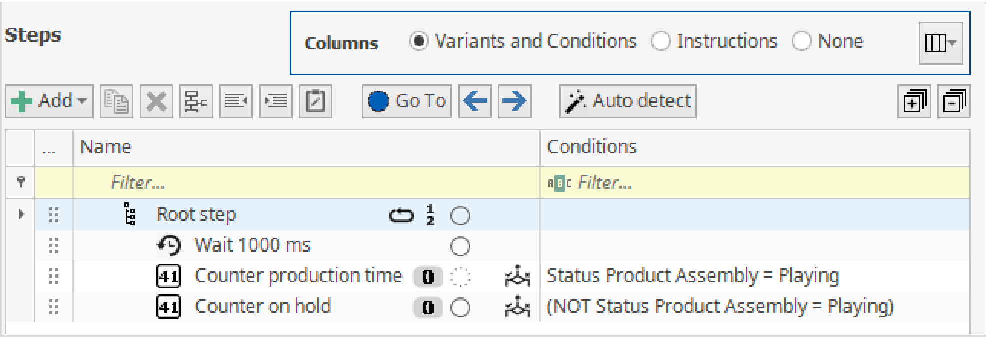
    Note
    When you link a condition to a composite step, it is automatically applies to all of its child steps as well. This is not always obvious because it is not shown for every specific child step in the user interface. If it ever occurs that a step is not executed and you can't find out why, double-check if none of its parent steps are linked to a condition that prevents it from being executed.

In the example below, we have configured a process that runs every second containing two counters:

- One for counting the time that the system is in production
- One for counting the time that the system is on hold

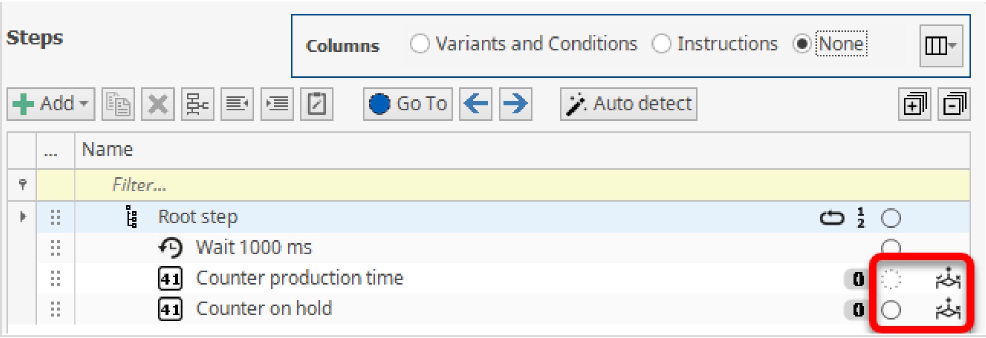
We have linked:

- the step **Counter production time** to the condition **Status Product Assembly = Playing** with **Expected State = True.**.
  It will only be executed if the value for the variable **Status Product Assembly** is equal to **Playing**.
  Every time the process runs (so in this case every second), the system validates the condition. If it is true, the step is executed and the counter increases with 1, otherwise the step is not executed and the counter remains at its current value.
- the step **Counter on hold** to the condition **Status Product Assembly = Playing** with **Expected State = False**.
  It will only be executed if the value for the variable **Status Product Assembly** is not equal to **Playing**.
  Every time the process runs (so in this case every second), the system validates the condition. If it is true, the step is executed and the counter increases with 1, otherwise the step is not executed and the counter remains at its current value.



When the **Conditions** column is not shown, you can still see which steps are linked to one or more condition(s) and therefore will not always be executed. They are marked by a icon.

You can also see which steps that are linked to a condition will be executed in the current situation (icon), and which will be skipped (icon).

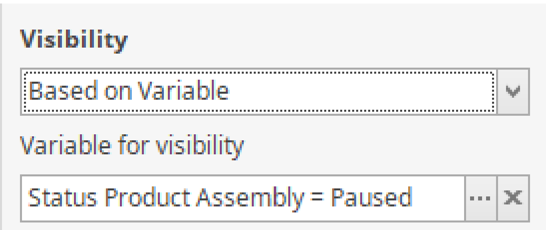


Using condition variables

For every condition you add, a variable will be created automatically as well, that has the condition's current state as value.

You can use these condition variables, for example to define the visibility of a visualization.

To do so, in the **Visibility** property select **Based on Variable**, and in the **Variable for visibility** select the condition variable of your choice. In the example below when the condition **Status Product Assembly = Paused** is in state True.



Managing conditions

To manage variants, click the **Conditions** button in the **Project Global** group in the ribbon.

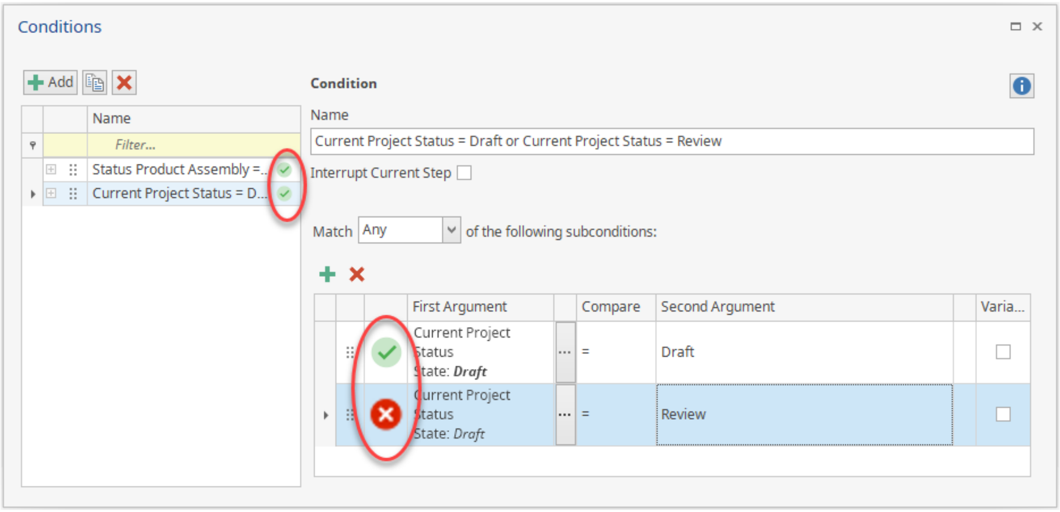
The **Conditions** window appears.

Select a condition and click to duplicate it.

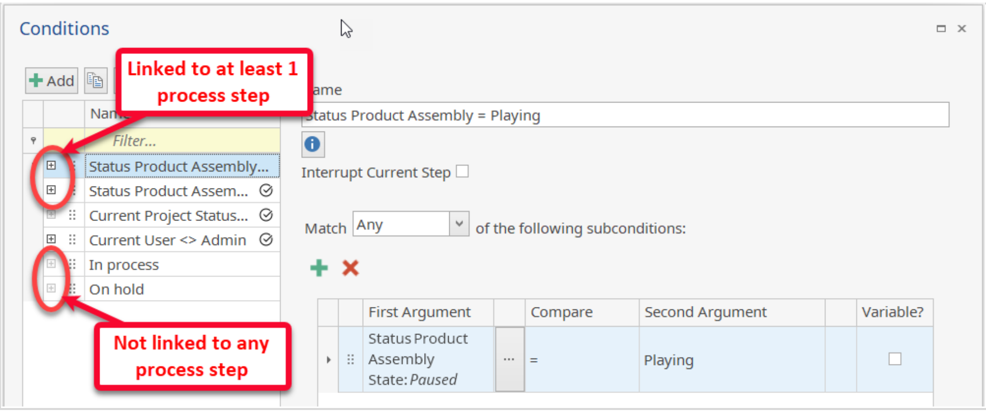
Select a condition and click to delete it.

Conditions and subconditions that are currently true are indicated with a icon.

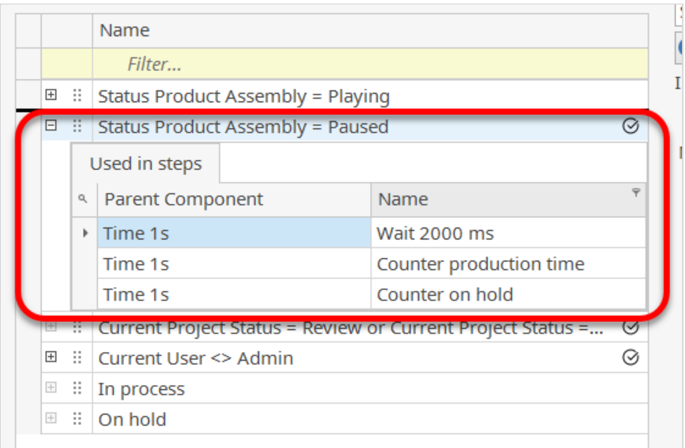
Conditions and subconditions that are currently true are indicated with a icon.



Conditions that are linked to one or more process steps are indicated with a ⊞ icon. For conditions that are not linked to any process step, the ⊞ icon is greyed out.



Click the ⊞ icon to see to which process steps the condition is currently linked.



5.8 Working with tasks

What?

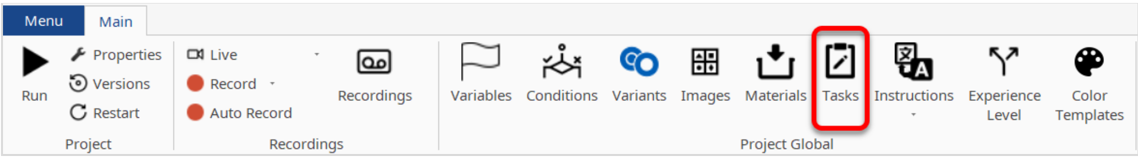
Tasks allow you to easily re-use a step or a group of steps across multiple processes.

Whenever you make changes to a task, those changes will automatically be applied at every location where you have used the task.

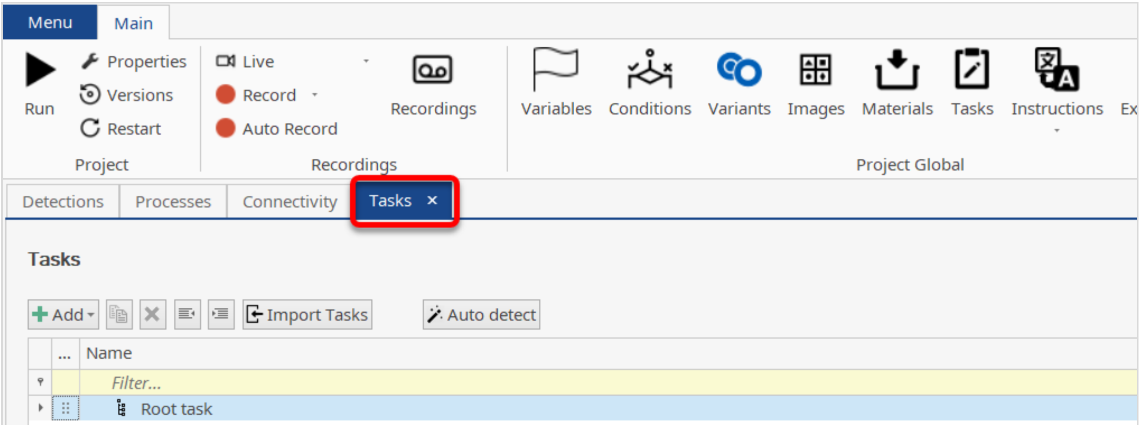
Adding tasks

To add a task, execute the steps below:

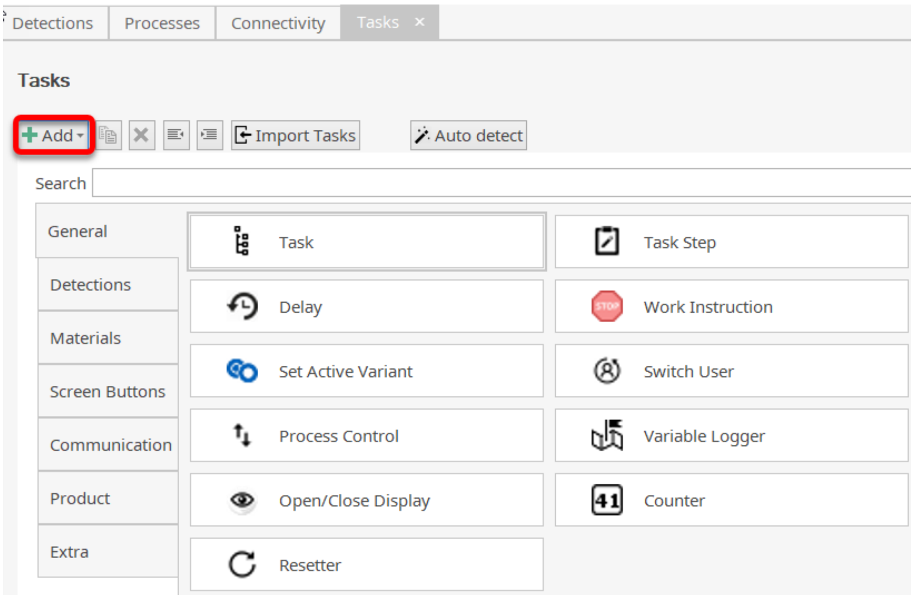
1. In the ribbon, in the **Project Global** group, click the **Tasks** button.



The **Tasks** tab is added to the list of tabs.



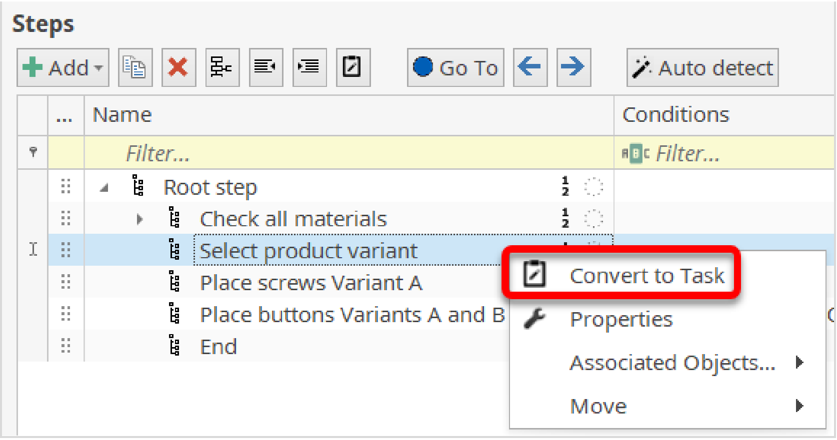
2. To add a task, click the **Add** button and select the process type you want to add.



3. Organize your process steps in the same way you would in a normal process.
   For more information, refer to section **5.4.3 Organizing your process steps** on page 112.

   Note

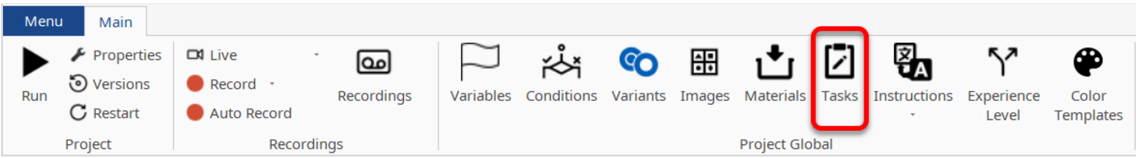
   *Alternatively, you can also create a task based upon a composite step in an existing process. To do so, right-click the step and select **Convert to Task**.*



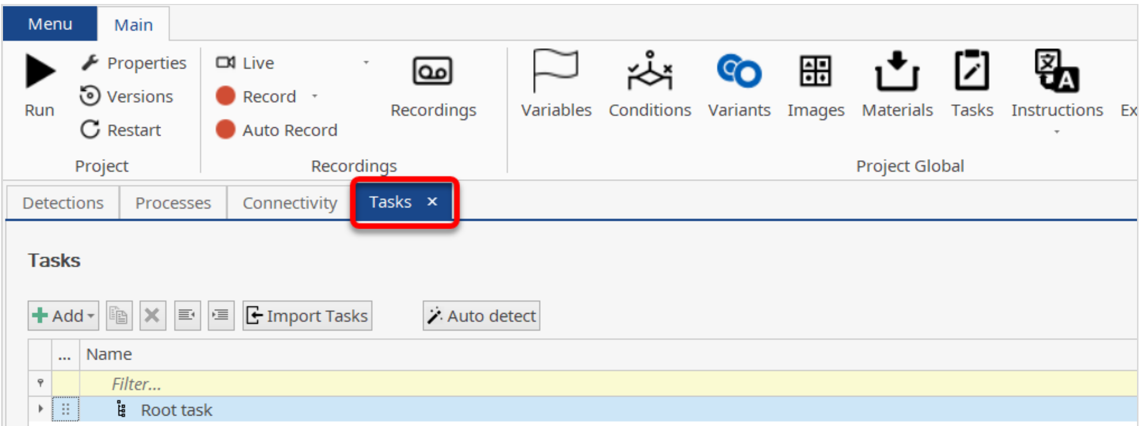
Importing tasks from another project

You can also add tasks to your project by importing them from another project. To do so, execute the steps below:

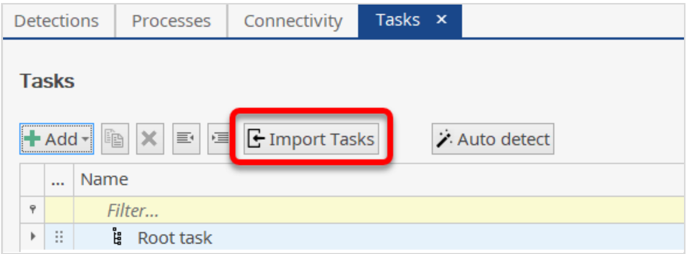
1. In the ribbon, in the **Project Global** group, click the **Tasks** button.



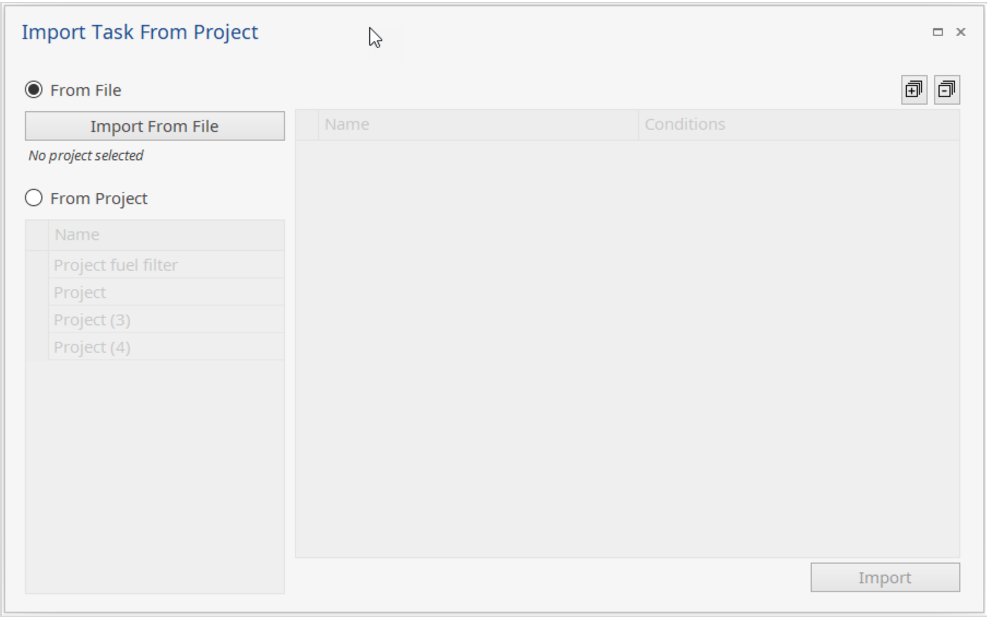
   The **Tasks** tab is added to the list of tabs.



2. Click the **Import Tasks** button.



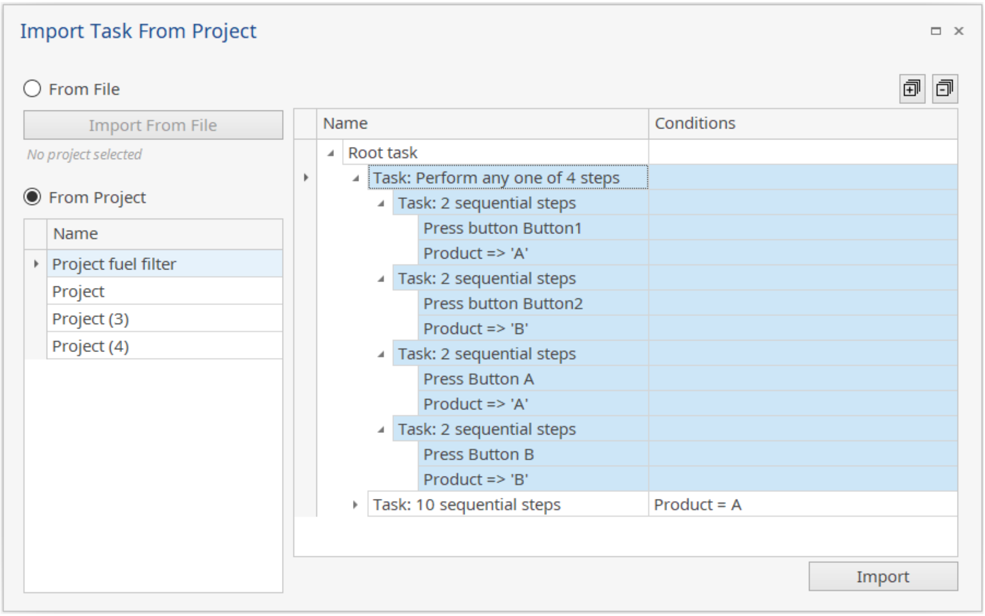
The **Import Task From Project** window opens.



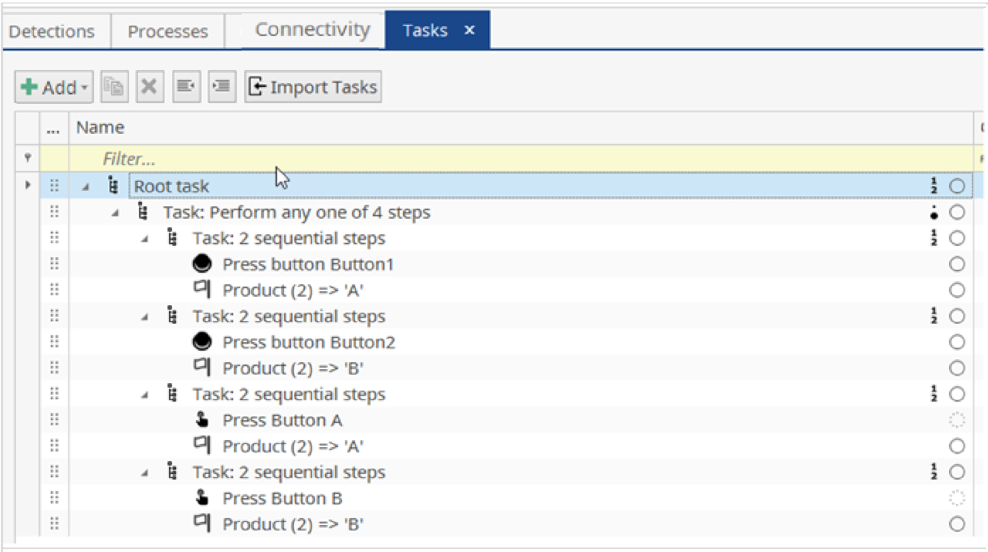
3. To import tasks from a project on your workstation, select the **From Project** radio button and select the project of your choice.
   To import tasks from a project file:
   - Select the **From File** radio button
   - Click the **Import From File** button
   - Select the .hef file of your choice and click **Open**.

   Note
   *To be able to import tasks from a project file, you should export the project first. For more information, refer to section **3.3 Managing projects** on page 34.*

4. Select the task(s) you want to import into your project.



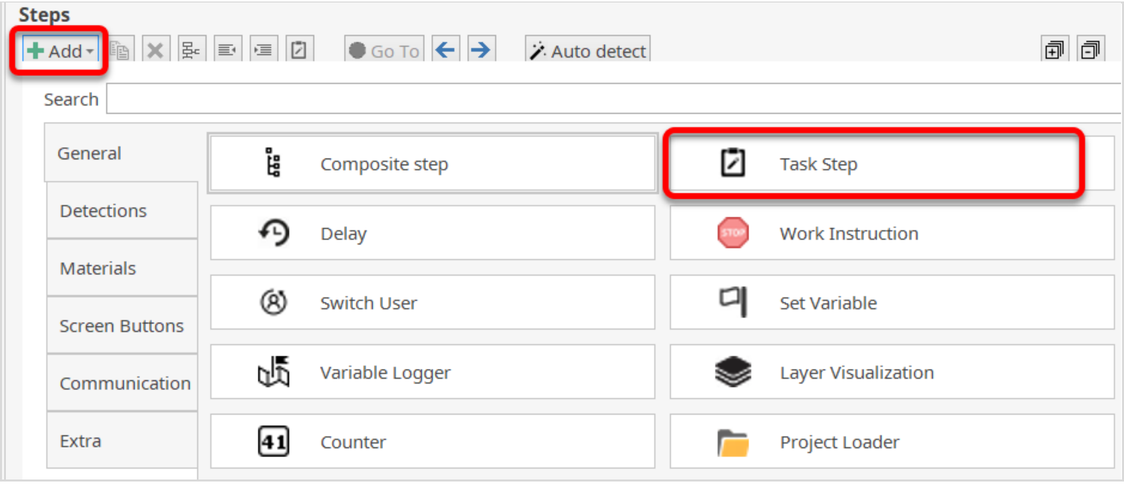
5. Click **Import**.
   The task(s) you selected are imported into your project.



Using tasks in your process

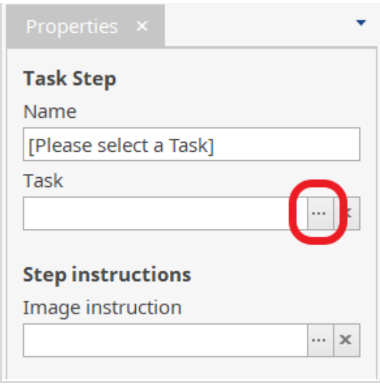
To use a task in your process, execute the steps below:

1. Navigate to the **Processes** tab.
2. In the **Steps** section, click the **New** button and on the **General** tab, select **Task Step**.

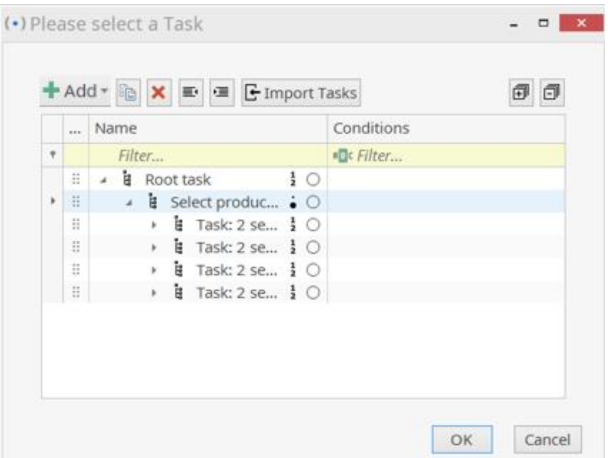


A task step is added to your list of process steps.

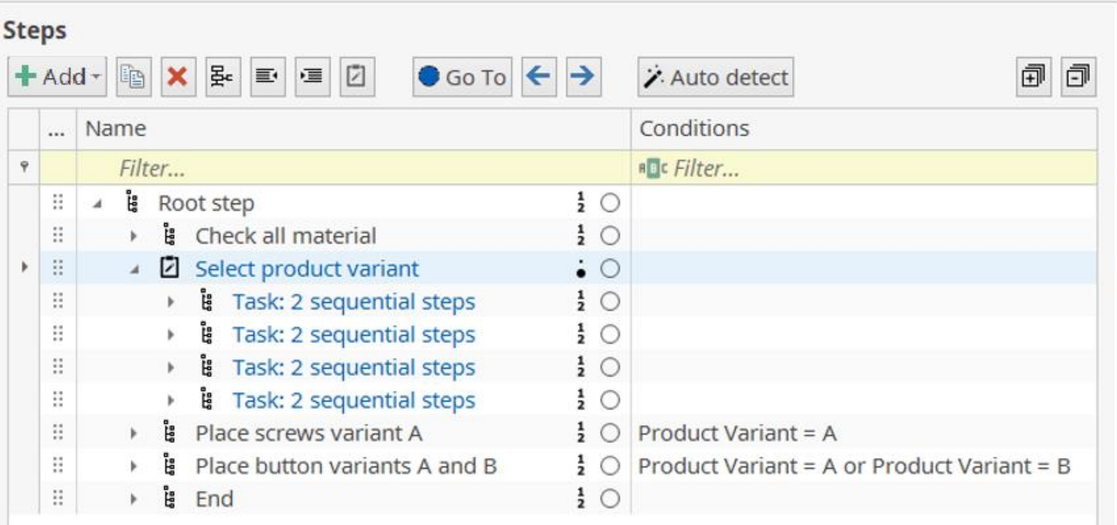
3. Select the task step in your list of process steps, and in the **Properties** pane, click the **...** button behind the **Task** field.



4. Select a task and click **OK**.

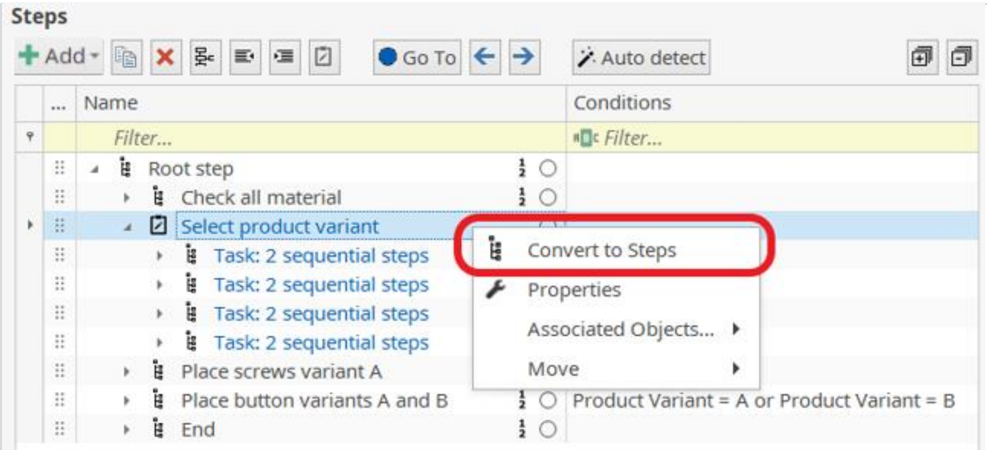


The task step is added to the list of process steps. It is rendered in blue and is read-only, which means you cannot make any changes to the steps. If you want to make any changes, you need to update the task itself. However, be aware that your changes will be applied at every location where you have used the task step.



Note

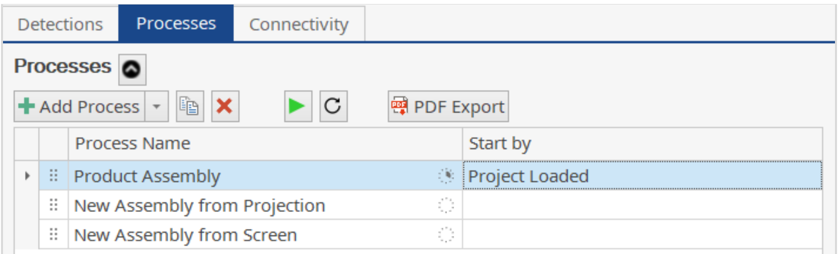
*If you want to make any changes that only apply for that particular process, you can unlink the task step from its task. To do so, right-click the task step and select **Convert to Steps**. Any future changes to the task will no longer be applied in this process.*



5.9 Managing processes

What?

After having set up all of your processes, you can manage them on the **Processes** tab.



How to manage processes

On the **Processes** tab, you can:

- Select a process and click to duplicate it.
- Select a process and click to delete it.
- Select a process and click to pause it or to play it
- Select a process and click to reset it.
  The process will start from the first step again.
- Select a process and click the **PDF export** button to create a pdf document containing all process steps with text and image instructions. If your project contains variants, a separate pdf document will be generated for every variant, as well as a pdf document that includes all steps of all variants.
  For every text and image visualization in your project, you can indicate if you want to show it in the pdf export or not.
  After the generation of the PDF exports, a dialog will appear with all the documents that have been created.

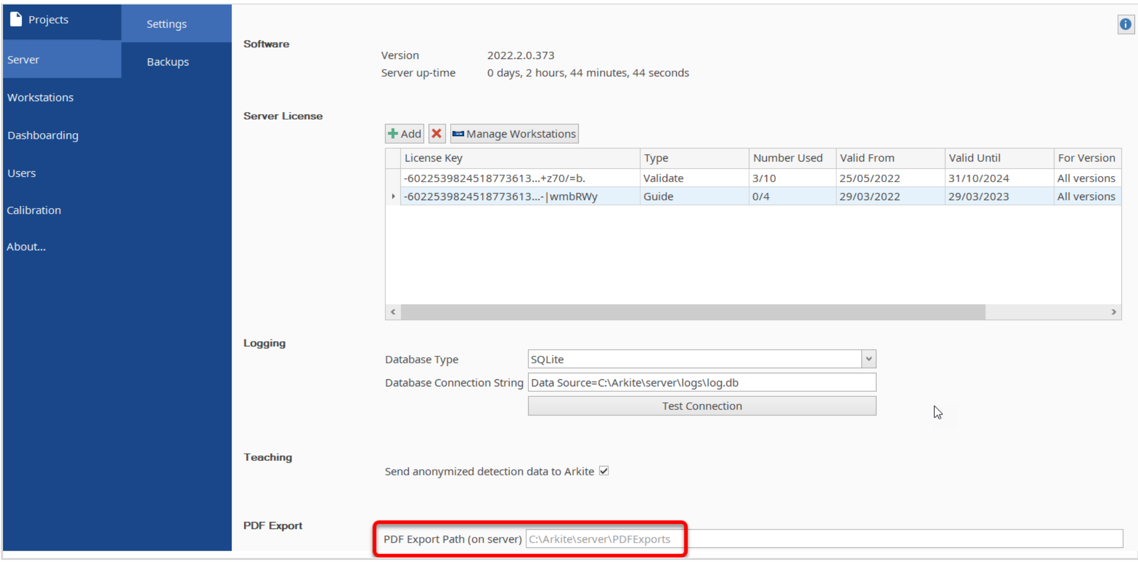
- To view a document, click the hyperlink for that document
- To download a document to your workstation, click the button in front of the document.

- To download all documents at once, click the **Download all PDFs to workstation** button. A ZIP file containing all PDFs will be downloaded.

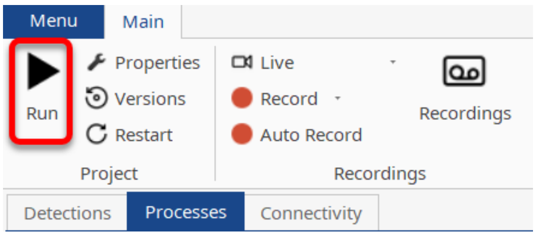
Note - PDF Export Settings

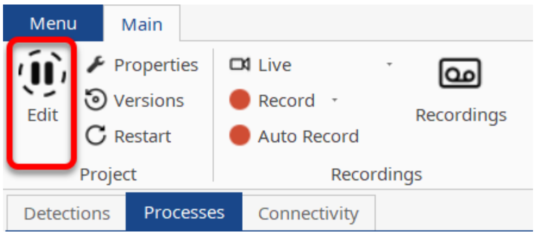
The pdf document is created in a folder on the server. By default, this is C:\Arkite\server\PDFExports. To change this folder:

- *Click the **Menu** tab.*
- *Click **Server** and then **Settings**.*
- *Under **PDF Export**, change the **PDF Export Path (on server)***



On the ribbon you can switch between **Run** and **Edit** mode.





When you make any changes to your project, it automatically switches to **Edit** mode.

When a project is (re)loaded or a process is started, it automatically switches to **Run** mode.

In **EDIT** mode:

- Processes are not running
- Projection of currently selected component or step is visible
  This makes it possible to finetune projections on the spot.

In **RUN** mode:

- Processes are running
- Projection of currently active step is visible

Note

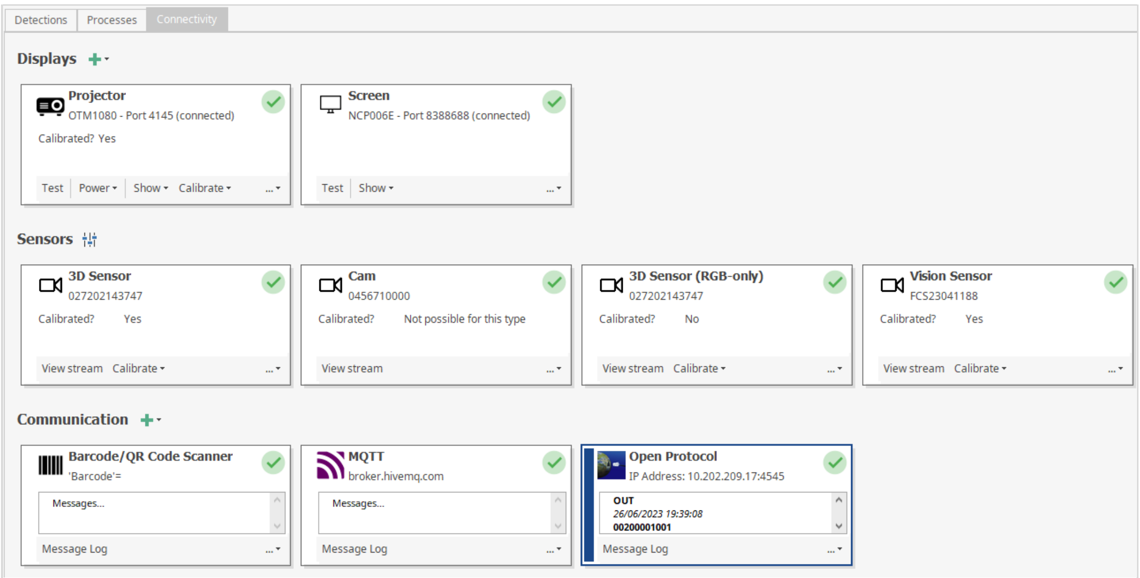
*On an editor, i.e. a workstation without a valid **Guide**, **Validate** or **Vision** license, you will be able to run a project for a maximum of 15 minutes. After that, it will automatically revert to **Edit** mode.*

6. Connectivity

6.1 About connectivity

The Arkite software allows to connect a variety of peripheral devices:

- Displays, i.e. projectors and (touch)screens that are connected.
- Sensors, i.e. the Arkite 3D sensor, the Arkite Vision sensor (when using a **Vision** license) and, if applicable, an RGB camera or webcam that is connected
- Communication modules, enabling the system to communicate to external devices.



6.2 Displays

6.2.1 What?

Once your project has been configured, the Arkite software will be able to provide your operators with real-time picking and assembly instructions through Augmented Reality. These instructions are provided on one or more displays.

6.2.2 Display types

Two types of displays can be distinguished:

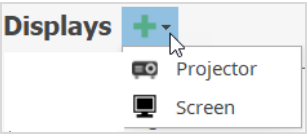
- **Projector**:
  One or more projectors are mounted above the workbench. These projectors enable you to show instructions on the workbench at the time the operator needs them.
- **Screen**:
  A (touch)screen can be connected to the workstation. On the screen, information and instructions can be displayed, as well as buttons to interact with the operator.

6.2.3 Adding a display

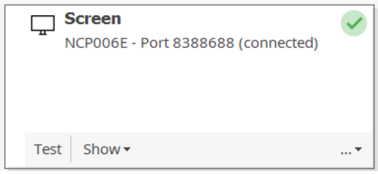
Steps

To add a display, execute the steps below:

1. Navigate to the **Connectivity** tab.
2. Under **Displays**, click the **Add** button and select the display type you want to add.



A new display card is added.



3. In the **Properties** pane, configure the properties for the display.

Projector Properties

Note

*It is also possible to communicate with a PJLink supported projector by adding a **Projector** process step to your process. The PJLink protocol uses an ethernet connection. It can for example turn the projector on or off, or switch the input source.*

Property	Description
Name	Name of the projector. ***Note*** *Next to the **Name** label, the object ID for the projector is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Auto Show	When checked, this display will be shown when the project is opened.
Background Color	The display's background color. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section **5.5.5 Managing visualization or DMX light colors** on page 130.*
Resolution	Select the resolution for the display. ***Note*** *When you have created the project before calibrating, a **Projector** display is added with the screen resolution of the device you are working on (e.g. your laptop). This may not match with the actual projector resolution, used during calibration. This will cause projections to be faulty. Update the **Resolution** property to the correct value to fix this.*
Display Device	Select the display device on which the projector will be shown.
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Projector IP address	The IP Address of the connected PJLink supported projector.
Port	The Port of the connected PJLink supported projector. PJLink has a default port of 4352.

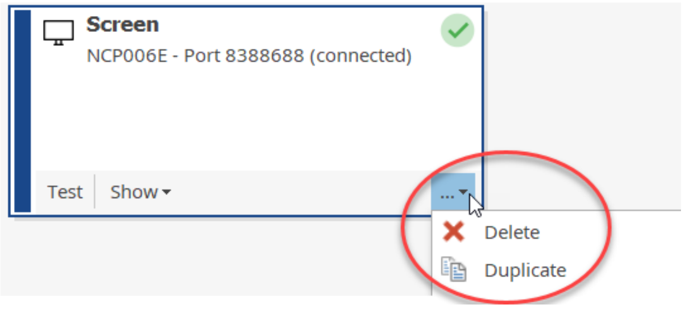
Screen properties

Property	Description
Name	Name of the screen. ***Note*** *Next to the **Name** label, the object ID for the screen is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Auto Show	When checked, this display will be shown when the project is opened.
Background Color	The display's background color. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section **5.5.5 Managing visualization or DMX light colors** on page 130.*
Resolution	Select the resolution for the display.

Property	Description
Display Device	Select the display device on which the screen will be shown.

6.2.4 Managing displays

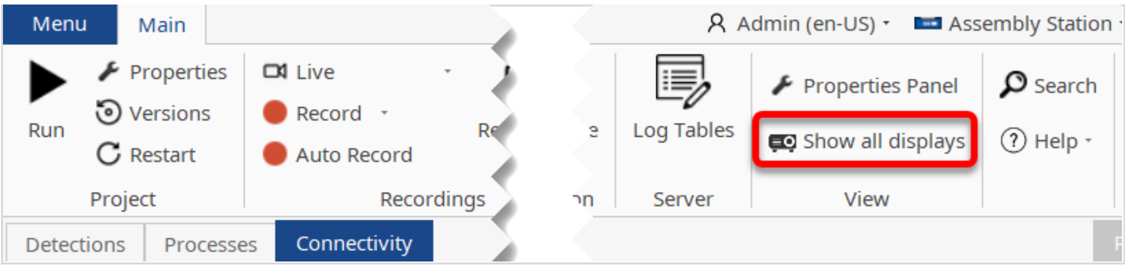
- To duplicate a display, select it, click the **...** button in the lower-right corner of the card and select **Duplicate**.
- To delete a display, select it, click the **...** button in the lower-right corner of the card and select **Delete**.



- Select a display and click **Show** to immediately display it.

  Tip
  If you're not working at the workbench location, you can choose to show the display on the screen of your pc.

  Note
  *You can show all displays at any time by clicking the **Show All Displays** button in the ribbon.*



- Select a display and click **Test** to verify which display is connected. The display will be bright yellow for 5 seconds to help you differentiate it from other displays.
- To turn a display on or off, select it, click **Power** and select **Turn on** or **Turn off**.

6.3 Communication modules

6.3.1 What?

The Arkite software allows to configure a variety of communication modules, enabling the system to communicate to external devices.

> ***Note***
>
> *A lot of communication modules use the workstation's ethernet ports. Always make sure the workstation's IP address and the IP address of the device you're trying to communicate with, are within the same range. Remember that the workstation has 2 separate network cards, each with its own settings.*

6.3.2 Communication module types

Barcode / QR Code Scanner

The Barcode / QR Code Scanner communication module is used to connect barcode and QR code scanners to the workstation through USB. Such scanners are widely used for product recognition. They convert a barcode or QR code to a specified combination of letters and numbers, or in other characters. This communication module can also be used as a keyboard logger or to connect an NFC reader, a foot pedal or hardware button. In the table below, you can find the properties you can configure for this communication module type.

Note

- *A barcode scanner technically works in the same way as a keyboard. If the scanned barcode appears incorrectly, please verify whether the Windows keyboard layout matches with the scanner.*
- *The Arkite software expects each barcode to end with a carriage return/line feed suffix. Make sure to correctly configure this in your barcode scanner's settings.*

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Variable for barcode / QR code	A variable that contains the last incoming text message from the scanner.

COM (serial port)

The COM communication module allows to establish serial communication with an external device. In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Receive Variable	The variable that will be used to store the received messages.
COM port	The COM port on which the external device is connected.
Baud Rate	The baud rate of the serial connection. Check the manual of the connected device to know the Baud Rate.
Data bits	The information to transmit, usually one of the values below: – 5 for baudot code – 7 for true ASCII – 8 for most kinds of data since this is the size of a byte
Stop bits	The stop bits allow the receiving signal hardware to detect the end of a character and to resynchronize with the character stream.
Parity	The party is a method of detecting errors in transmission: – **None**: no parity bit is sent at all. – **Odd**: Counts the given set of bits with value 1, sets the parity bit so the count is odd. – **Even**: Counts the given set of bits with value 1, sets the parity bit so the count is even. – **Mark**: Set the parity bit to 1. – **Space**: Set the parity bit to 0.
Handshake	Handshaking is an automated process for negotiation between two entities before communication is possible. – **None:** No control is used for the handshake – **XOnXOff:** the XON/XOFF software control protocol is used. – **RTS:** Request to send hardware flow control is used. – **RTS/XOnXOff:** Both RTS and XOnXOff software controls are used.
Receive Append Until Symbol	If unchecked, every incoming message is used to update the selected variable and spark subsequent components. If checked, incoming data is concatenated until a newline character appears, even if this data is spread across multiple messages.
Receive New Line Symbol	Select the new line symbol to use for receiving: – CR LF (\r\n): carriage return + line feed – CR (\r): carriage return – LF (\n): line feed – Custom: specify a new line symbol of your choice

Property	Description
Send Append Symbol	If checked, the 'Send New Line Symbol' is appended at the end of a message when sending.
Send New Line Symbol	Select the new line symbol to use for sending. – CR LF (\r\n): carriage return + line feed – CR (\r): carriage return – LF (\n): line feed – Custom: specify a new line symbol of your choice

Database

The Database communication module is used to communicate with a MySQL, oracle, SQL Server or sqlite database. A variable is used to represent a query which will be executed against the database. If the query is a select statement, you can either choose to store the first row of the result in a number of variables or to store all rows as a JSON array. In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Variable for last message type	A variable storing the message type of the last message: – 'INFO' – 'WARNING' – 'ERROR'
Variable for last message	A variable storing the last message for this communication module
Database Type	Database type. Currently supported types are: – SQL Server – Oracle Database – Sqlite – MySQL
Server	The name or URL of the server running the database.
User	The user who is used to query the database. This user should have the proper rights to access the table defined below.
Password	The password used to authenticate the user with. ***Note*** *You can click the button to make the password you've entered visible and verify if you haven't made any typing mistakes.*

Property	Description
Database	The name of the database on the server. ***Note*** *In case of SQLite, this should be the path to the database file on the server's file system.* *For example:* ***C:\Arkite\server\mydatabase.db***

DMX

The DMX communication module is used to connect DMX lights through an Enttec Open DMX USB interface (SKU 70303). You can then link your detections to a DMX light, and use the DMX light for operator guidance instead of a projection. Alternatively, you can also control a DMX light through a **Send Communication** step. In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the* ***Name*** *label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter* ***17 The Arkite API*** *on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section* ***12.3 Changing the logging database and the logging database type*** *on page 221.*
DMX device	The serial number of the Enttec Open DMX USB interface that is linked to the communication module. By default, the first unlinked controller that is found will be linked automatically.
Configure Lights	Click this button to configure different lights on the connected DMX device. For more information, refer to section **6.3.4 Configuring lights for a DMX communication module** on page 179.

Epson Label Printer

The Epson Label Printer communication module is used to communicate with Epson label printers through the EPOS protocol over ethernet. It is possible to print logo's, variables, barcodes, QR codes. The module has been tested with the Epson TM-T88VI. In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the* ***Name*** *label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter* ***17 The Arkite API*** *on page 250.*

Property	Description
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section* ***12.3 Changing the logging database and the logging database type*** *on page 221.*
IP address	The IP Address of the connected Epson label printer. ***Note*** *Epson label printers have "192.168.192.168" as the default IP address.*
Device ID	The id of the target printer to be used. The default value is **local_printer**.
Time Out (ms)	The time out specified in the printer. The default value is **60000**.

Ethernet (TCP/IP)

The Ethernet communication module is used to send and receive messages over the TCP/IP protocol.

It can also be used to send messages to or receive messages from a multicast group. In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the* ***Name*** *label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter* ***17 The Arkite API*** *on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section* ***12.3 Changing the logging database and the logging database type*** *on page 221.*
Receive Variable	The variable that will be used to store the received messages.
Variable for Remote IP	The variable will be used to store the Remote IP Address from which incoming messages are sent.
Variable for Remote Port	The variable will be used to store the Remote Port from which incoming messages are sent.
Multicast	Check if you want to send a multicast message over UDP.
Multicast IP Address	The Multicast Address where you want to send the message to.
Multicast Time To Live	Mechanism which limits the lifespan or lifetime of data in a computer or network: – **1** Restricted to the same host. Won't be output by any interface. – **1** Restricted to the same subnet. Won't be forwarded by a router. – **< 32** Restricted to the same site, organization or department. – **< 64** Restricted to the same region. – **< 128** Restricted to the same continent. – **< 255** Unrestricted in scope. Global.
Remote IP Address	The IP address you want to send a message to.
Remote Port	The TCP-port on the remote device.

Property	Description
Local Port	The TCP-port to listen on. The Remote device should send its messages to this port.

Inter-Workstation

The Inter-Workstation communication module is used to send events between two workstations. The module needs to be configured on both workstations to be able to communicate properly. In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Remote Workstation IP Address	The IP address of the ‘other’ workstation. You can receive messages and send messages to this workstation, using the same communication module. Click the **Find workstations** button to find workstations on the same network. ***Tip*** *If you want to know the IP address(es) of a workstation, press CTRL-ALT-A on that workstation.*
Local Port	The TCP/IP port to use on your workstation.
Remote Port	The TCP/IP port to use on the remote workstation.

Important

*For the Inter-Workstation communication to work, the communication module needs to be configured on both workstations. The **Local Port** on the first workstation needs to be the same one as the **Remote Port** on the second workstation, and vice versa.*

HTTP

The HTTP communication module is used to communicate with external web services through the get, post, put, patch and delete request methods. You can send data from your workstation to the web service, and process JSON responses. In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*

Property	Description
Log Table	The table name in the logging database in which the information below will be logged for each HTTP request: – the URL – the Response Code – the first 1000 characters of the message ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Use OAuth2 Authentication	If checked, OAuth2 Authentication will be used. Additional properties will appear (see below). Based on these properties, an OAuth2 Access Token will be generated, which will be sent in the headers with each request.
OAuth2 Client ID	The OAuth2 Client ID
OAuth2 Client Secret	The OAuth2 Client Secret
OAuth2 Token URL	The URL which will be used to generate the Access Token using the client id and secret

Modbus

The Modbus communication module is used to communicate (read/write) with an external device through the Modbus TCP/IP protocol. In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Modbus Device IP address	The IP address of the Modbus device.
Port	The port of the Modbus device. The default port for Modbus protocol is 502.
Polling rate	The interval in milliseconds between readings. The default polling rate is 250ms.
Slave Address	The address of the specific slave device. Typical value is 1.

MQTT

The MQTT communication module is used to communicate through MQTT (Message Queue Telemetry Transport), a lightweight and flexible network protocol that supports asynchronous communication between parties. The MQTT protocol defines two types of network entities: a message broker and a number of clients. An MQTT broker is a server that receives all messages from the clients and then routes the messages to the appropriate destination clients. Your workstation(s) will act as MQTT client(s). In the table below, you can find the properties you can configure for this communication module type.

<table>
<tr><th>Property</th><th>Description</th></tr>
<tr><td>Name</td><td>Name of the communication module

Note
Next to the Name label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter 17 The Arkite API on page 250.</td></tr>
<tr><td>Log Table</td><td>The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged.

Note
The logging database is configured in the server settings. For more information, refer to section 12.3 Changing the logging database and the logging database type on page 221.</td></tr>
<tr><td>Variables</td><td>Variables that will be synced with the MQTT broker. For each variable, the information below can be configured:
<table>
<tr><td>Variable</td><td>Name of the variable</td></tr>
<tr><td>Topic</td><td>Information on the broker and the clients are organized in a hierarchy of topics.
Topic on the MQTT broker where the value will be read from or written to.</td></tr>
<tr><td>Publish/Subscribe</td><td>– Publish:
any change to the state of the variable will be sent to the MQTT broker and stored in the relevant topic.
– Subscribe:
any change to the topic value on the MQTT broker will be stored in the variable’s state.</td></tr>
<tr><td>State</td><td>The current value of the variable.</td></tr>
</table></td></tr>
<tr><td>Broker Address</td><td>The IP address of the MQTT broker.</td></tr>
<tr><td>Port</td><td>The port on which to connect to the MQTT broker. The default port for the MQTT protocol is 1883, or 8883 when TLS is used.</td></tr>
<tr><td>Use credentials</td><td>If checked, a username and password are used to authenticate when connecting to the MQTT broker.</td></tr>
<tr><td>Username</td><td>The user to connect to the MQTT broker.</td></tr>
<tr><td>Password</td><td>The password used to authenticate the user with.

Note
You can click the button to make the password you’ve entered visible and verify if you haven’t made any typing mistakes.</td></tr>
<tr><td>Use TLS</td><td>If checked, the communication is encrypted using the TLS (Transport Layer Security) protocol.</td></tr>
</table>

Property	Description
Client ID	ID that is used to identify the workstation when connecting to the MQTT broker. It is advised to choose something meaningful to you, e.g. the workstation name (assembly station, packing table...) In case you don't fill in a client ID, Arkite will randomly generate one. ***Note*** *In some MQTT brokers, you need to configure the client IDs that can connect. Those MQTT brokers won't accept randomly generated client IDs. Please consult your MQTT broker's documentation for more information.*

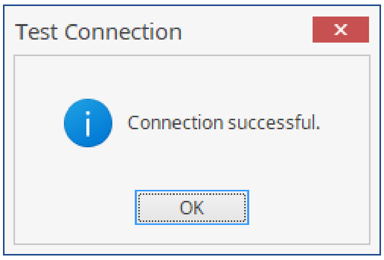
OPC UA

The OPC UA communication module is used to communicate using a Open Platform Communications (OPC) standard. More specifically, this module can synchronize variables on a workstation with a table on an OPC UA server containing a column for the variable name and one for the variable state.

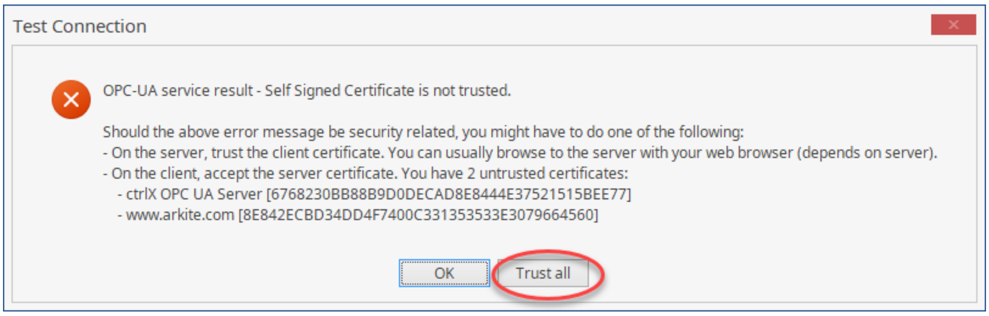
In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Server	The name of the OPC UA Server to communicate with. Should be in the format **opc.tcp://[ip address]:[port number].**
User Name	The user who is used to access the table on the OPC UA server.
Password	The password used to authenticate the user with. ***Note*** *You can click the button to make the password you've entered visible and verify if you haven't made any typing mistakes.*

When connecting to an OPC UA server, you can test your connection by clicking the **Test Connection** button. If communication is successful, you will get the popup below:



In case there are any untrusted server certificates on the client, there will be a **Trust All** button. Click the button to trust all untrusted server certificates.



Alternatively, you can also cut the files from **C:\Arkite\client\OPC\CertificateStores\RejectedCertificates** and paste them to **C:\Arkite\client\OPC\CertificateStores\UA Applications\certs**

You might also have trust the client certificate on the server. You can usually browse to the server with your web browser.

Open Protocol

The Open Protocol communication module uses the TCP/IP protocol to control an external Open Protocol compliant device (called a controller), for example a smart tool.

> ***Note***
>
> *This module can only be used to communicate with devices on the same network.*

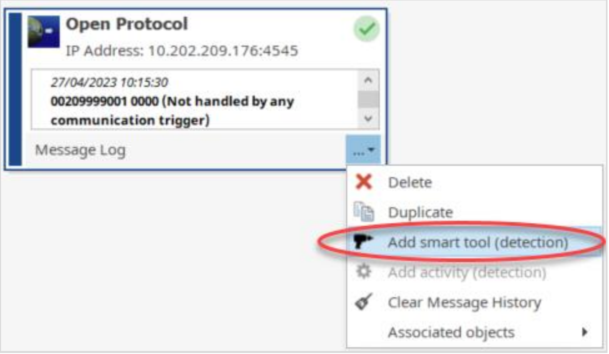
In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Automatic Smart Tool Settings	If this checkbox is enabled, the required variables for communication with a smart tool will be generated and configured.
IP Address	The IP address of the remove device, for example: 10.0.11.12.
Device Type	– When using Atlas Copco’s MicroTorque Controller MTF6000, select **MTFocus 6000**. – For all other devices, select **Generic**.
Port	The TCP/IP port to use. Default for open protocol communication is 4545.
Start communication revision	When an open protocol communication actuator is first sparked, a ‘start communication’ message will be sent to the controller. This message contains a 3 digit revision. For most controllers, this revision is ‘001’. Check the manual of your controller to check for divergent revisions.
End communication revision	When the module is paused, an ‘end communication’ message will be sent to the controller. This message contains a 3 digit revision. For most controllers, this revision is ‘001’. Check the manual of your controller to check for divergent revisions.

Property	Description
Stop String Type	Every message will be ended by a special 'ending character'. The default value is null (null character). Other types include a CRLF (carriage return line feed) and #.
Log 'Keep Alive' messages	Every incoming, outgoing and exception message will be logged. However, to keep the connection with the controller open, 'keep alive' messages will be sent (and received) every ten seconds. In most situations, you don't want to see these messages cluttering your message stream. However, if you would like to see these messages, please check this checkbox.
Open Protocol Device Test	Before an open protocol device can be used, a parameter set must always be selected. Provide a parameter set in the text input field (typically consisting of three digits, e.g. 000, 001, 002....) and click **Set Parameter**. Once you have done that: – Click the **Enable Tool** button to enable the tool. – Click the **Disable Tool** button to disable the tool. By clicking these buttons, the communication towards the tool can be tested.

Once you've configured the open protocol module, you can set up a smart tool that uses the module:

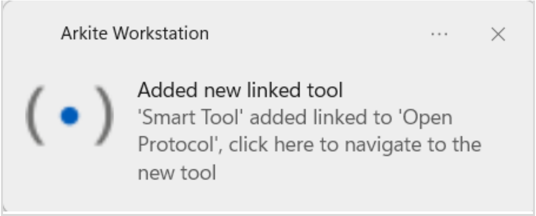
1. Select the card for the **Open Protocol** module, click the **...** button in the lower-right corner of the card and select **Add Smart Tool (detection)**.



Note

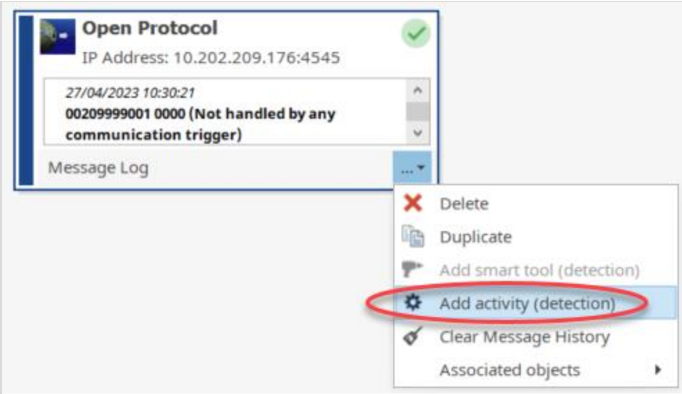
*Only __one__ tool can be linked to a single **Open Protocol** module. To use multiple tools, create multiple **Open Protocol** modules.*

2. A smart tool linked to the **Open Protocol** module is created on the **Detections** tab. Click the toast message in the lower-right corner of the screen to navigate to it.

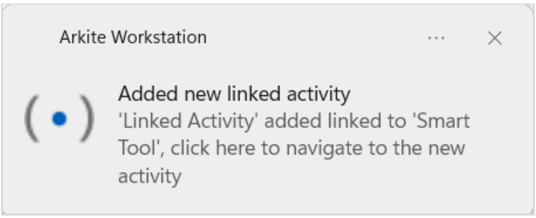


3. (optional) When using a **Validate** or **Vision** license, teach the tool as described in section **4.5 Teaching and finetuning detections (Validate and Vision license only)** on page 64.
   This way the system can detect when the tool is taken away or placed back.

4. On the **Connectivity** tab, select the card for the **Open Protocol** module, click the **...** button in the lower-right corner of the card and select **Add Activity (detection)**.



5. An activity linked to the smart tool you set up in step 5 is created on the **Detections** tab. Click the toast message in the lower-right corner of the screen to navigate to it.



6. (optional) When using a **Validate** or **Vision** license, teach the activity as described in section **Teach an activity** on page 70.

   This way the system can detect when the tool is being used at the desired location.

7. Add more activities as needed.

S7 PLC

The S7 PLC communication module allows to communicate with an external S7 compliant PLC device. It makes use of the TCP/IP network protocol.

Supported S7 devices are: Siemens Logo 0BA8, S7-200, S7 200 Smart, S7-300, S7-400, S7-1200, S7-1500.

> ***Note***
>
> *This module can only be used to communicate with devices on the same network.*

In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*

Property	Description	
Variables	Variables that will be synced with the PLC. For each variable, the information below can be configured:	
	Variable	Name of the variable
	DB	Data Block in the PLC where the value will be read from or written to.
	Data Type	Can be any of the following: – **Bit:** Boolean value. Can be TRUE or FALSE, 0 or 1 – **Byte**: bit string of 8 bits. – **Word:** bit string of 16 bits. – **DWord:** bit string of 32 bits. – **Int:** signed integer with a length of 16 bits. Values can be -32 768 to +32 767. – **DInt:** signed integer with a length of 32 bits. Values can be -2 147 483 648 to +2 147 483 647. – **Real:** signed floating number with a length of 32 bits. The precision defined by the IEEE754 standard is stored. Additionally specified decimals are rounded off. – **LReal:** signed floating number with a length of 64 bits. The precision defined by the IEEE754 standard is stored. Additionally specified decimals are rounded off. – **String:** ASCII character. – **S7String:** ASCII character string incl. special characters. The length of the string can be defined, and can contain up to 254 characters. – **S7WString:** Unicode character string. The default length is 254 characters, but you can define a length up to 16382 characters. – **Timer:** contains a timer operation – **Counter:** contains a counter operation – **DateTime:** stores date and time information in BCD format (year-month-day-hour:minute:second:millisecond) – **DateTimeLong:** stores date and time information in a predefined structure (Year-Month-Day-Hour:Minute:Second.Nanoseconds)
	Offset	Relative address in the datablock where the value will be read from or written to. Addresses in a datablock are defined by a byte and a bit. – For the data type **Bit**, use the format **byte.bit.** For example, "10.4" will read from or write to bit 4 of byte 10. – For the data types **String** or **WString**, use the pattern **byte.bit[length].** For example, "20.0[100]" will read or write 100 bytes starting from byte 20. If you don't provide a length, the default length of the data type will be used. – For all other data types, use the format **byte.** For example: "54" will start reading from or writing to byte 54. The number of bytes that will be written or read depends on the length of the data type. In case the data type is dint, the length is 4 bits. Bytes 54 to 57 will be read or written.
	I/O	Indicates if the variable will be written to or read from the PLC.
	State	The current value of the variable. Accepted variables depend on the selected **Data Type**.

Property	Description
	Log Message: If checked, the sent or received data will be logged in the Message History. ***Note*** *For more detailed information, consult the documentation of your PLC programming application.*
S7 Type	The type of the connected S7 device.
IP Address	The IP address of the connected S7 device.
Rack number	The rack number of the connected S7 device. Default value is 0.
Slot number	The slot number of the connected S7 device. Default value for S7-200, S7-300 and S7-400 is 2, S7-1200 and S7-1500 is 1.
Polling rate	The interval in ms with which the variables will be synced with the PLC.

Variable Storage

The Variable Storage communication module is a communication module used to communicate with a database. More specifically, this module can synchronize variables with a table containing a column for the variable name and one for the variable state. In the table below, you can find the properties you can configure for this communication module type.

Property	Description
Name	Name of the communication module ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Log Table	The table name in the logging database in which all incoming and outgoing messages and exceptions will be logged. ***Note*** *The logging database is configured in the server settings. For more information, refer to section **12.3 Changing the logging database and the logging database type** on page 221.*
Database Type	Database type. Currently supported types are: – SQL Server – Oracle Database – Sqlite – MySQL
Server	The name or URL of the server running the database.
User	The user who is used to query the database. This user should have the proper rights to access the table defined below.
Password	The password used to authenticate the user with. ***Note*** *You can click the button to make the password you've entered visible and verify if you haven't made any typing mistakes.*

Property	Description
Database	The name of the database on the server. ***Note*** *In case of SQLite, this should be the path to the database file on the servers file system.* *For example:* ***C:\Arkite\server\mydatabase.db***
Table	The table on the database, which will contain the event values.
Variable Name Column	The column of the table containing the name of the variable(s)
Variable State Column	The column of the table containing the state of the variable(s)

6.3.3 Adding a communication module

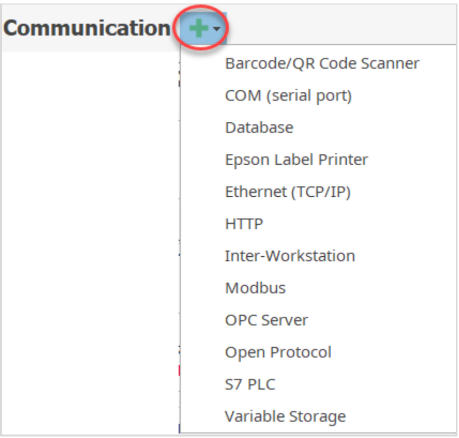
What?

Before you can start using a communication module, you need to add and configure it in the Arkite software.

Steps

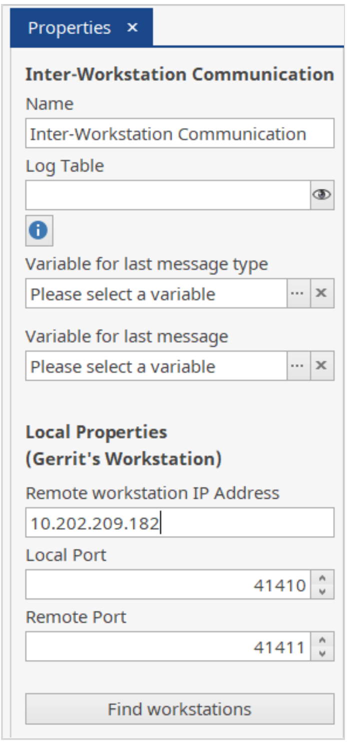
To add a communication module, execute the following steps:

1. In the **Connectivity** tab under **Communication**, click the **Add** button and select the communication module you want to add.



A new card for the communication module is added to the list of communication modules.

2. In the **Properties** pane, configure the properties for the communication module.



Note

*The properties that need to be configured depend on the communication module type you have added. For an overview of each property and its meaning, refer to section **6.3.2 Communication module types** on page 164.*

Result

The communication module has been configured and will immediately start running. Data will be received through the communication module. The status icon in the upper right corner of the card indicates if the module is correctly configured and running.

For the communication module to send data, you need to add a **process step**, defining which data to send. You can also configure a **trigger** for the communication module to start a process when receiving data.

6.3.4 Configuring lights for a DMX communication module

What

After you have connected a DMX device through a DMX communication module, you can configure different lights on that device by defining which channels to use.

After configuring a light, you can control it by linking it to a detection. This way, you can use the DMX light for operator guidance instead of a projection. Alternatively, you can also control the light through a **Send Communication** process step.

Steps

1. In the **Configure Lights** window, click the + button.
   A light is added with default name **L1**.
2. Edit the properties for the DMX Light:

Property	Description
Name	Name of the DMX light ***Note*** *Next to the **Name** label, the object ID for the communication module is shown. You can copy it to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.*
Start Address	The first channel on the DMX device that will be used to control the light. By default, the first unused channel will be selected.
Number of channels	The number of channels on the DMX device that will be used to control the light. By default, 4 channels will be used.
Channel Mapping	Depending on the **Start Address** and the **Number of Channels**, a table is shown with a row for each channel. Select what each channel will be used for in the configuration of an RGB color value and light intensity (Dimmer). For channels that should not be used, select **None**.
Max Dimmer Range	Configure the value that corresponds to the light's maximum light intensity. By default 255.

Test your configuration

After configuring your light, you can use each channel using the **R, G** and **B** toggle buttons. The light will emit the corresponding color until the button is pressed again, the window is closed or another light is selected To test the **Dimmer** channel, use the slider next to the toggle buttons. Use the **Off** button to completely turn off the light manually.

Reset Lights

To reset all lights linked to the DMX module to their default value, typically off, click the Reset **Lights** button on the communication module.

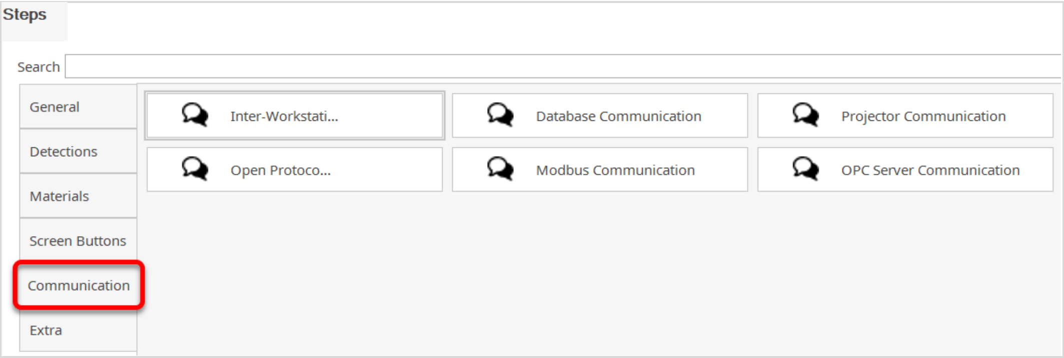
6.3.5 Configuring a process step for the communication module to send data

What?

In order to enable the communication module to send data, you need to configure a process step, defining when which data will be sent.

Steps

1. Navigate to the **Processes** tab.
2. Under **Steps**, click the **Add** button.
3. Select the **Communication** subtab and click the communication module you want to use to send data.



Note

When using Open Protocol communication, you can select the action you want to execute from the context menu. The correct message will then be preconfigured. The available options are:

- *Enable Tool*
- *Disable Tool*
- *Select Parameter*
- *Select Job*
- *Other...: this will allow you to specify your own message*

*Before an open protocol device can be used, a parameter set must always be selected. If the tool does not react after sending an **Enable Tool** message, make sure a parameter set is selected. Send a **Select Parameter** message first.*

Note

*When using DMX communication, you can select the light you want to control from the context menu. In most use cases, you will control a DMX light by linking it to a detection. For more information, refer to section **4.4 Adding detections** on page 60.*

Note

*Apart from the communication modules you have added in the **Connectivity** tab, you can also select a **Projector** display that is available in the **Displays** section of the **Connectivity** tab here.*

4. Configure the properties for the process step.
   The properties that need to be configured depend on the communication module type. You can find an overview of properties per communication module type below.

   - **COM (serial port)**

Property	Description
Variable	The variable that contains the message you would like to send

   - **Database**

Property	Description
Query	Enter the query you want to run on the database. To use the value of a variable as query, click the icon and select the variable containing (part of) the query. Alternatively, you can also type the variable name between double braces, i.e. *{{Variable name}}*. You can also use a variable within a variable. For example, if the value of a variable **Index** is 01, then *{{Variable{{Index}}}}* will resolve into the value of the variable **Variable01**. When communicating with an SQL server database, you can also call stored procedures in that database. ***Tip*** *For long texts, click the icon to enlarge the text box.*
Result	– **All rows:** all rows that are returned are stored in a JSON array, with a maximum of 10 000 rows. – **First row:** only the first row in the result is taken into account. For every column in the result, a variable with the same name is created or updated. In the **Prefix for column variables** property, you can optionally specify a prefix for those variable names.
Variable for result	Only available when the **All Rows** option is selected. The variable in which the result of the query will be stored as a JSON array.
Prefix for column variables	Only available when the **First Row Only** option is selected. Names of newly created variables (as a result of a 'select' query) will be prefixed with this value

   - **DMX**

Property	Description
DMX Light	The DMX light you want to control.
Configure Lights	Click this button to configure different lights on the connected DMX device. For more information, refer to section **6.3.4 Configuring lights for a DMX communication module** on page 179.
Color	The color you want the DMX light to show. This color will be shown until the color of the light is changed or until the light is put out by another **Send Communication** step or when a detection linked to the light is activated. ***Note*** *You can either set a fixed color, select two alternating colors, define the color based on the value of a variable or define the color based on a template. For more information, refer to section **5.5.5 Managing visualization or DMX light colors** on page 130.*

Note

*In most use cases, you will control a DMX light by linking it to a detection. For more information, refer to section **4.4 Adding detections** on page 60. If you are adding a **Send Communication** step to control a light that is linked to a detection, a warning message will be shown.*

- **Epson Label Printer**

Property	Description
Variables	The variables that are going to be printed. Check the **All Variables** checkbox to print all variables.
Print Variable Name	When checked, the variable names are printed together with the current state.
Divider Text	The text between the variable name and its current state.
Print Logo at Top	When checked, the logo is printed, specified as key1=32 key2=32 defined in the printer.
Print Barcode	When checked, a barcode is printed.
Barcode Variable	The variable state that will be encoded and printed as a barcode. ***Note*** *The barcode is restricted up to 15 numeric characters.*
Print QR	When checked, a QR code is printed.
QR Variable	The variable state that will be encoded and printed as a QR code (QR Code Model 2). Valid characters include alphanumeric, _, -, brackets... ***Note*** *Some special characters are not supported.*
Cut at Bottom	When checked, the printer will cut the label at the bottom.

- **Ethernet (TCP/IP)**

Property	Description
Variable	The variable that contains the message you would like to send
Add the name of the event ('name:value')	When checked, the variable name and value are both sent.
Pad	When checked, messages will always have the same length (number of characters).
Pad To Size	The length of the message (number of characters). If the message is not long enough, zeros will be added to the message. Only applicable when 'pad' is checked.
Use CRLF	When checked, the message ends with a Carriage Return Line Feed.
Wait for response when sending	When checked, the TCP/IP connection remains open after sending the message until a response from the remote device is received. When unchecked, the TCP/IP connection is closed immediately after sending the message.
Response Variable	The variable where the response from the remote device is stored when the **Wait for response when sending** property is checked.

- ### Inter-Workstation

Property	Description
Variables	When the variable(s) you indicate here are received from the other workstation, the process will be started. Check the **All Variables** checkbox to start the process when receiving any variable.

- ### HTTP

Property	Description
URL	The URL of the web service to communicate with, for example: http://myservice.com/logger To use the value of a variable in the URL, click the icon and select the variable containing (part of) the URL. Alternatively, you can also type the variable name between double braces, i.e. *{{Variable name}}.* ***Tip*** *For long texts, click the icon to enlarge the text box.*
Variables for Query	The variables you specify here are encoded in the query part of the URL. If the **Variables for Query** list contains one element **variableName** the request could look like this: http://myservice/logger?variableName=variableValue
Resulting URL	The requests that results from the contents of the **URL** and **Variable for Query** properties.
Request Type	The HTTP request type. You can select one of the values below: – **Get:** this method is used to retrieve the representation of a resource – **Post:** this method is used to create new resources – **Put:** this method is typically used to update capabilities of a resource. The request body contains the newly-updated representation of the original resource. – **Patch:** this method is typically used to modify capabilities of a resource. The request body contains the changes to the resource, not the complete resource. – **Delete**: this method is used to delete a resource identified by a URI
Body Content Type	Only visible when the **Request Type** is **Post, Put** or **Patch**. Determines the content type of the body. There are currently two options: – **Raw** (=default option): You can specify the body of the request in the **Raw Body Content** property. – **FormData:** The variable names and their current states you specify in the **Variables for Form Data** property will be sent in FormData format.
Variables for Form Data	When the **Body Content Type** is **FormData**, the variables you specify here are encoded in the body of the request in FormData format.
Content Type	When **Body Content Type** is **Raw**, the content type of the raw data should be specified. By default, this is ‘application/json’

Property	Description
Raw Body Content	When **Body Content Type** is **Raw,** you can compose the body of the request here. To enter the value of a variable in the body content, click the icon and select the variable you want to insert. Alternatively, you can also type the variable name between double braces, i.e. *{{Variable name}}*. ***Tip*** *For long texts, click the icon to enlarge the text box.*
Extra Headers	Pass additional information by including extra headers in your request. An HTTP header consists of its case-insensitive name followed by a colon (:), then by its value. In the **Extra Headers** property you can add one or more names with their corresponding values. You don't need to add the colons. To use the value of a variable in the **Value** column, click the icon and select the variable you want to insert. Alternatively, you can also type the variable name between double braces, i.e. *{{Variable name}}*.
Response Variable	The variable in which the response data will be stored.

- ### Modbus

Property	Description
Address Type	The object type of the write action. It is possible to write: – **Coils**: read-write entity with a size of 1 bit (ON/OFF) – **Holding registers:** read-write entity with a size of 16 bits
Address	The address of the object which will be changed to the specified state.
State to set	Only available when the **Address Type** is **coil**. The state to which the object will be changed. The state can be either on or off.
Value	Only available when the **Address Type** is **holding register**. The value to which the object will be changed. The value can be a number between 0 and 65 535.

▪ Open Protocol

Property	Description
Header	Depending on the option you have selected in the context menu when adding the process step, the header will be prefilled with the Open Protocol message that is required to complete the action of your choice: – Enable Tool: 00200043 (Generic) / 0023022500100000000 008 (MTFocus 6000) – Disable Tool: 00200042 (Generic) / 0023022400100000000 008 (MTFocus 6000) – Select Parameter: 00230018001 (Generic and MTFocus 6000) – Select Job: 00220038001 (Generic and MTFocus 6000) – Other...: the header remains empty, you can specify a message yourself. ***Note*** *Before an open protocol device can be used, a parameter set must always be selected. If the tool does not react after sending an **Enable Tool** message, make sure a parameter set is selected. Send a **Select Parameter** message first.* To have a full overview of all message types that are supported by Open Protocol, you can consult the **Open Protocol specification documents** embedded below. Generic MTFocus 6000 Open Protocol messages consist of 20 digits (for example. 00200001000000000000 for communication start). – Digit 1 – 4: Length – Digit 5 – 8: MID – Digit 9 – 11: Revision – Digit 12: No ack flag – Digit 13 – 14: Station ID – Digit 15 – 16: Spindle ID – Digit 17 – 20: Spare ***Note*** *If the header is less than 20 digits, it will be padded with spaces until it contains 20 digits before being sent.* *If the header is more than 20 digits, the excess will be considered data.*
Variable for data	The current state of this variable will be appended to the header. Together with the 'ending character', this will be the full message to be sent to the controller. ***Note*** *The ending character is defined by the property **Stop String Type** configured in the communication module. Refer to section **Open Protocol** on 173 for more information.*

▪ OPC UA

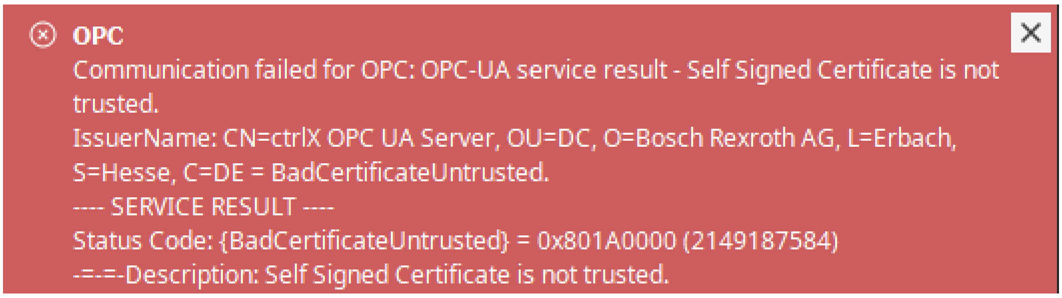
Property	Description
Type	The action you want to do on the OPC UA Server: – Read – Write
Variable for result	When **reading** a node/item, the value of this variable will be updated with the value from node/item on the OPC UA server.
Variable to send	When **writing** to a node/item, the value of the node/item on the OPC UA server will be updated with value of this variable.

Property	Description
Node id	The data item inside the OPC UA server. Click the **Find Node** button to select a node.

Note

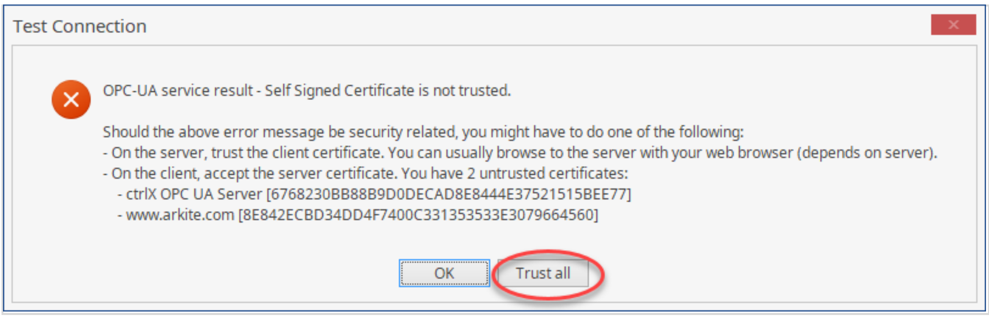
- ***Trusting Server Certificates on the client***

  If you are getting a popup or a notification about missing or non-trusted certificates, you are missing OPC Server Certificates.



  *On the **Connectivity** tab, select the **OPC UA** communication module and click the **Test Connection** button in the **Properties** pane.*

  *Click the **Trust All** button to trust all server certificates on the client.*



  *Alternatively, you can also cut the files from **C:\Arkite\client\OPC\CertificateStores\RejectedCertificates** and paste them to **C:\Arkite\client\OPC\CertificateStores\UA Applications\certs***

- ***Trusting Client certificates on server***

  *If you get an exception **The server may have rejected the connection because it does not trust the client**, you should trust the client certificate on the server. You can usually browse to the server with your web browser.*

- **Projector**

Property	Description
Projector Command	The commands below can be sent to the projector: – **Power On:** turns on the projector – **Power Off:** turns off the projector – **Switch To VGA:** switches the active input of the projector to VGA – **Switch To HDMI1:** switches the active input of the projector to HDMI1 – **Switch To HDMI2:** switches the active input of the projector to HDMI2 – **Mute On:** mutes the video and audio of the projector – **Mute Off:** stops muting the video and audio of the projector – **Get Current State:** returns the (power) state of the projector – **Get Current Input:** returns the active input of the projector – **Get Error Code:** returns the error code of the projector (returns “000000” when there is no error) – **Get Lamp Hours:** returns the number of hours the projector has been active
Variable for answer	The variable in which the responses of the projector will be stored.

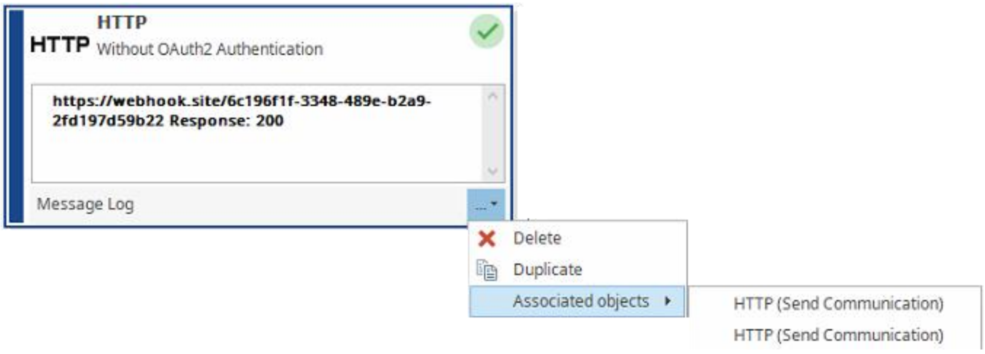
- **Variable Storage**

Property	Description
Variables	The variables that you want to fetch, send or remove to/from the database. Check the **All Variables** checkbox to insert/update/delete all variables.
Type	The action that will be performed on the database: – **Select:** Read variables from the database and set the state of the project variables to the values present in the database – **InsertOrUpdate:** Insert the project variables into the database, or, if they are already present in the database, update their value – **Remove:** Remove the specified variables from the database

Result

The communication module has been configured to send data.

On the **Connectivity** tab, the outgoing communication will be listed in the **Associated objects** for the communication module. Click the **...** button in the lower-right corner of the card and select **Associated objects.** The screenshot below shows an example for the **HTTP** communication module.



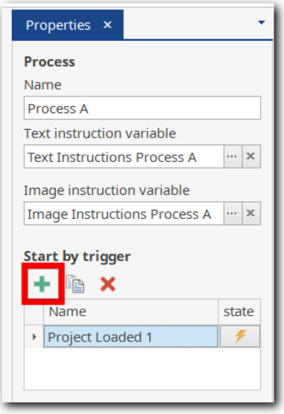
6.3.6 Configuring a trigger for the communication module to start a process when receiving data

What?

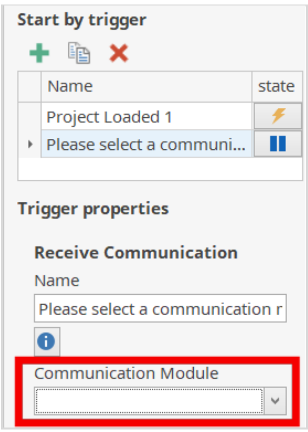
You can configure a process to start when data is received through the communication module you have added. In order to do so, you need to configure a communication trigger.

Steps

1. On the **Processes** tab, select the process you want to trigger by the receival of communication.
2. In the **Properties** pane, under **Start by trigger**, click the button.



3. Select **Receive Communication**.
4. Under **Trigger Properties**, select the module you want to configure as a trigger from the **Communication Module** dropdown list.



5. Configure the properties for the trigger.
   The properties that need to be configured depend on the communication module type. You can find an overview of properties per communication module type below.

   - **Inter-Workstation**

Property	Description
Variables	The variables that are going to be received. Check the **All Variables** checkbox to send all variables.

- **Modbus**

Property	Description
Variable	The variable that will be used to store the received messages.
Address Type	The object type of the read action. It is possible to read: – **Discrete inputs:** read-only entity with a size of 1 bit (ON/OFF) – **Coils**: read-write entity with a size of 1 bit (ON/OFF) – **Input registers:** read-only entity with a size of 16 bits (0-65 535) – **Holding registers:** read-write entity with a size of 16 bits (0-65 535)
Address	The address of the object from which will be read. The process will be started when a message is received from the address specified here, with the correct type.

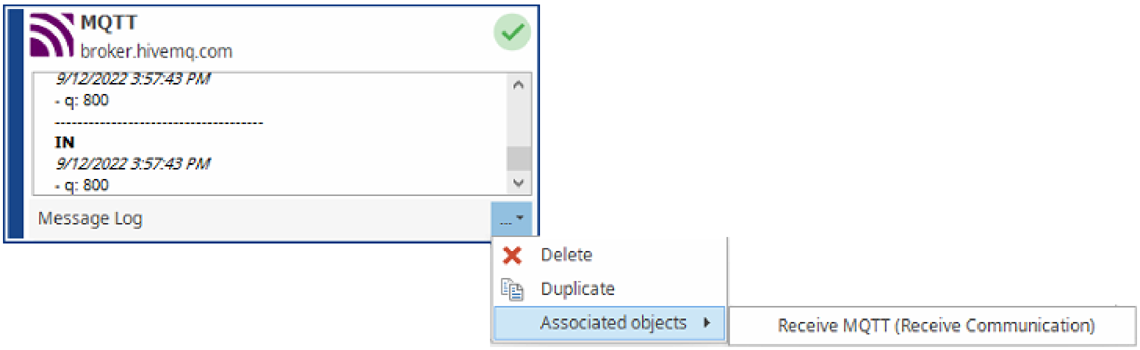
- **Open Protocol**

Property	Description
MID to filter on	The MID (which is character 5 to 8 of the message header) to filter the messages on. If this field is filled in, only messages with a matching MID will be captured and the process will only be started when the matching MID is received. If this field is left empty, all messages will be captured. The process will started whenever a message is received.
Variable	The variable that will be used to store the received messages.

Result

The communication module has been configured to receive data.

On the **Connectivity** tab, the incoming communication will be listed in the **Associated objects** for the communication module. Click the **...** button in the lower-right corner of the card and select **Associated objects.** The screenshot below shows an example for the **MQTT** communication module.



6.3.7 Following up on the communication that has been exchanged

What?

Once you have correctly set up the communication module and the associated triggers or actuators, you can follow up on the exchanged information.

How to follow up on exchanged information

On the **Connectivity** tab, the latest outgoing and incoming communication will be listed in the **Messages...** field on the card for the communication module.

Three types of messages can appear:

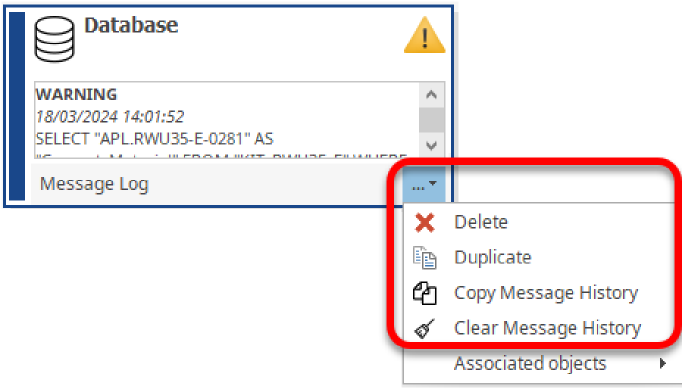
- **IN:** Incoming messages
- **OUT:** Outgoing messages
- **ERROR:** Error messages, specifying what is wrong

To view more messages, click the **Message Log** button.

6.3.8 Managing communication modules

- To duplicate a communication module, select it, click the **...** button in the lower-right corner of the card and select **Duplicate**.
- To delete a communication module, select it, click the **...** button in the lower-right corner of the card and select **Delete**.
- To copy a communication module's message history, select it, click the **...** button in the lower-right corner of the card and select **Copy Message History**.

  The message history is copied to your clipboard.
- To clear a communication module's message history, select it, click the **...** button in the lower-right corner of the card and select **Clear Message History**.



6.4 Sensors

What?

The **Sensors** section in the **Connectivity** tab manages the sensors that are used in the project. Whenever a sensor is connected, it is automatically added to the list.

The items in the list show the sensors that are used in the project.

On a typical workstation with a **Validate** license, two sensors appear: the Arkite 3D sensor, a time-of-flight sensor that has the ability to measure both distance and infrared, and an RGB camera for regular color images. Webcams are also supported: simply plug it into an available USB port, and it appears in the list.

On a typical workstation with a **Vision** license, a third sensor appears: the Arkite Vision sensor.

If sensors are at some point not available, they remain to be part of the project. For example, if a project is edited on a laptop, the sensors are not connected but remain part of the project. This allows to continue editing without having an actual sensor connected.

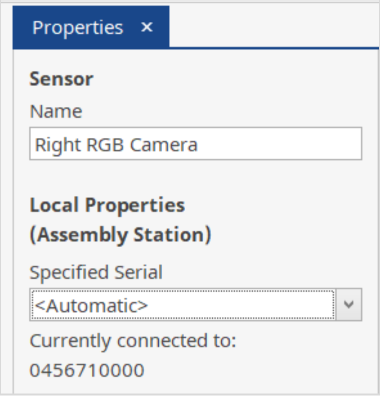
> ***Note***
> *The Arkite 3D and Arkite Vision sensos are detected by default when connected. If you are using the Arkite software with a different type of sensor, click the button to enable the relevant sensor type. Contact Arkite if you want to use a sensor type that is not in the list.*

Specifying Sensor Serial Number

If a project uses multiple sensors of the same type, you have to specify their serial number, so the software knows without ambiguity which connected sensor corresponds to which logical sensor of the project.

For example, if two webcams are used to make product snapshots, and one is called ‘Left RGB camera’ and the other is called ‘Right RGB camera’, you have to specify which serial number corresponds to the ‘Left RGB camera’, and do the same for the right camera. You have to specify this for every unit that uses the project, as the serial numbers of the respective sensors will be different. Whenever action is required, a warning appears in the status bar.

To specify the sensor serial number, select the sensor card. In the **Properties** pane, select the correct serial number in the **Specified Serial** dropdown list. In the **Name** field, you can specify a meaningful name for the sensor.

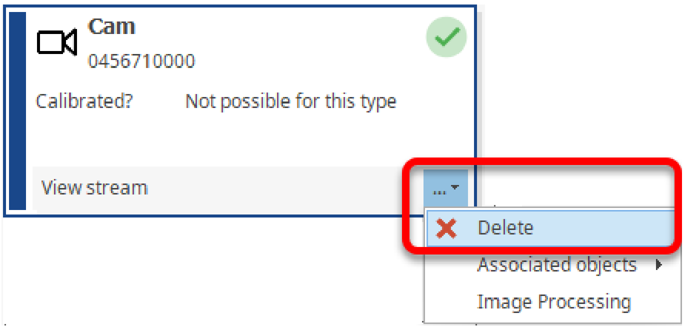


Note

In a typical case, where only one sensor of a certain type is used, there is no need to specify the serial numbers and the default ***<Automatic>*** *can be left untouched. The software will automatically use the attached sensor of the same type.*

Removing a sensor

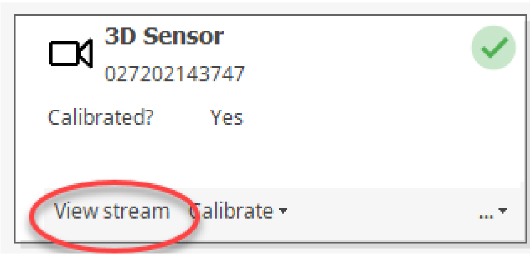
To remove a sensor, click the **...** button in the lower-right corner of the senor's card and select **Delete.**



An example use case is when an external webcam was required in a project was, which was at one point connected and hence a webcam sensor appeared in the list. If, later on, the webcam is no longer needed, it can be permanently removed.

Viewing the sensor stream

To view images from the sensor on your screen, click the **View Stream** button in the sensor card.



7. Experience levels

7.1 About experience levels

The **Experience level** functionality allows you to customize your project and change its behavior based on the operator's level of experience.

When the **Experience Level** functionality is disabled, the project behavior will be the same for all operators.

When the **Experience Level** functionality is enabled, by default three operator experience levels become available. Depending on the experience level, you can customize:

- Detection settings (Validate and Vision license only)
- Projection settings
- Display of visualizations

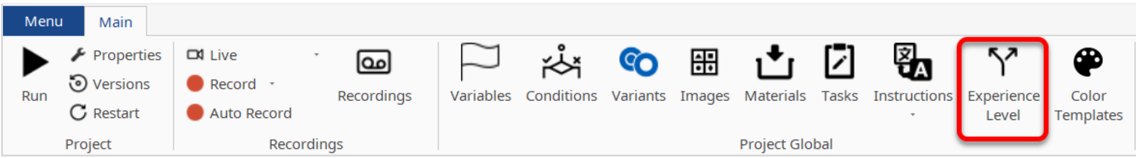
The experience level can either be set manually or it can be defined by the Arkite software, based on learning calculations.

7.2 Managing experience levels

Enabling experience levels

To enable the **Experience level** functionality, execute the steps below:

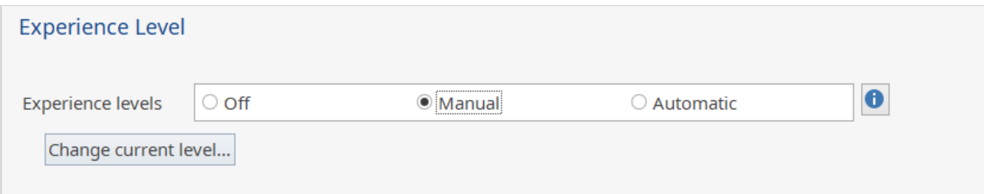
1. In the ribbon, in the **Project Global** group, click the **Experience Level** button.



   The **Experience Level** window opens.

2. Under **Experience levels**:

   - Select the **Manual** radio button to let the operator or the project administrator define the experience level.
   - Select the **Automatic** radio button to let the Arkite software define the experience level.



   The **Experience level** functionality is enabled. The functionality in the **Experience Level** window becomes available.

Available experience levels

By default, three experience levels are available:

- Level 1: Training
- Level 2: Basic
- Level 3: Advanced

After enabling the **Experience level** functionality, by default level 1 will be used.

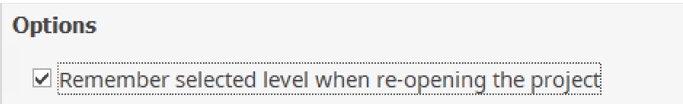
Modifying experience level names

To change the name for an experience level, update its label in the **Experience Level** window under **Level names.** The experience level name will be updated throughout the application.

Experience level options

In the **Experience Level** window under **Options**, you can specify whether:

- the selected experience level should be retained after restarting the project.
- the experience level should be reset to the default Level 1 every time the project is restarted.



7.3 Experience level settings

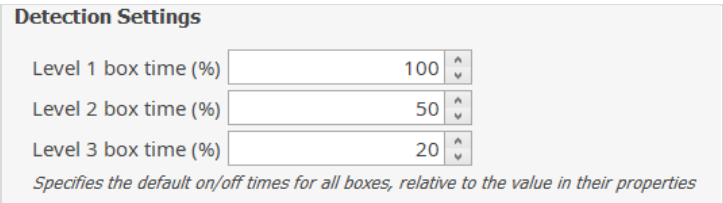
Detection settings (Validate and Vision license only)

For every detection in your project, there are time threshold properties that define for how long an object needs to be within (ON) or outside (OFF) the range defined in the fill rate properties to change detection state.

In the **Experience Level** window under **Detection Settings**, you can set a percentage of these time thresholds that will be used for every specific experience level.

By default, these percentages are set to:

- 100% for level 1
- 50% for level 2
- 20% for level 3



This means that for a detection with a time threshold of 40 ms, the following thresholds will be used for each specific level:

- 40 ms for level 1
- 20 ms for level 2
- 8 ms for level 3

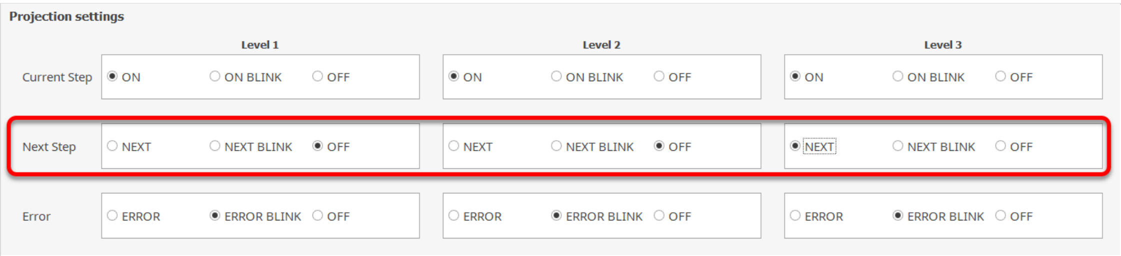
You can change the percentages for every level as required. It is also possible to overrule these default percentages for a specific detection. For more information, refer to section **4.5.4 Finetune a detection** on page 74.

Projection settings

In the **Experience Level** window under **Projection Settings**, you can customize the projection settings independently for each experience level.

The options are:

- For the current step in your process:
  - **ON:** a projection will be shown
  - **ON BLINK:** a projection will be shown and it will blink
  - **OFF:** no projection will be shown
- For the next step in your process:
  - **NEXT:** a projection will be shown
  - **NEXT BLINK:** a projection will be shown and it will blink
  - **OFF:** no projection will be shown
- For errors during your process (Validate and Vision license only):
  - **ERROR:** a projection will be shown
  - **ERROR BLINK:** a projection will be shown and it will blink
  - **OFF:** no projection will be shown



Example

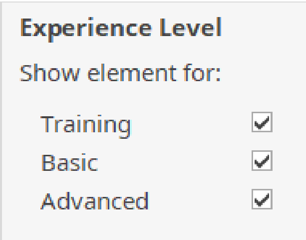
*In the screenshot above, a projection for the **Next Step** will be shown to users with **Level 3 (Advanced)**, so that they can anticipate. For users with **Level 1 (Training)** or **Level 2 (Basic)**, no projection will be shown because they might be confused by the additional projection.*

Note

*To change the colors of these projections, click the **Color Templates** button in the ribbon and select the **Detection colors** template. There you can update the colors for the projection settings mentioned above. For more information, refer to section **14.2 The Detection Colors template** on page 232.*

Display of visualizations

In the **Properties** of a visualization, there is a new **Experience Level** property.



This way, you can define whether the visualization will be displayed for a specific experience level or not. By default, every visualization is shown for all experience levels.

> ***Example***
>
> *You can choose to show text instructions to users with **Level 1 (Training)** or **Level 2 (Basic)** but hide them for users with **Level 3 (Advanced)**, since there is no real added value for those users.*

> ***Note***
>
> *For more information on the properties of a visualization, refer to sections **5.5.1 Visualization types** on page 116 and **5.5.2 Visualization properties** on page 117.*

7.4 Selecting the experience level

Auto-selection

When automatic experience levels have been selected, the Arkite software defines the operator's experience level based on learning calculations.

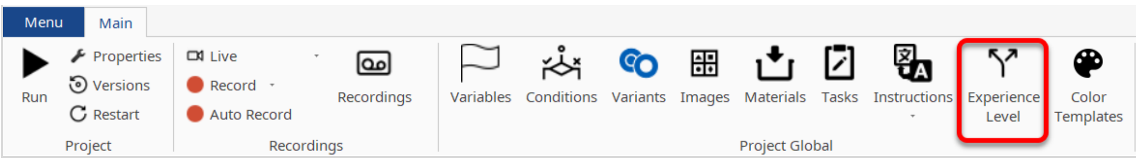
> ***Note***
>
> *The experience level can still be changed by the project administrator in the Arkite software, but after a few additional repetitions of the assembly process the value will be overwritten again with the one that has been calculated automatically by the Arkite software.*

Selection by the project administrator

When **manual** experience levels have been selected, the project administrator can change the experience level in the Arkite software.

To change the experience level in the Arkite software, execute the steps below:

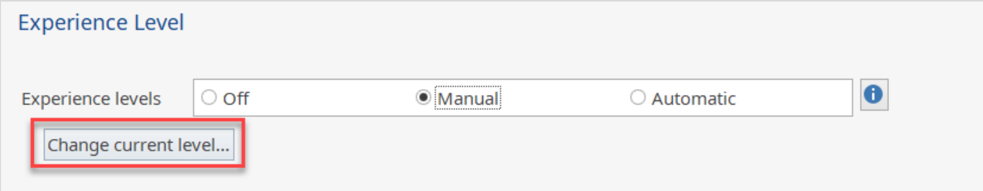
1. In the ribbon, in the **Project Global** group, click the **Experience Level** button.

   The **Experience Level** window opens.

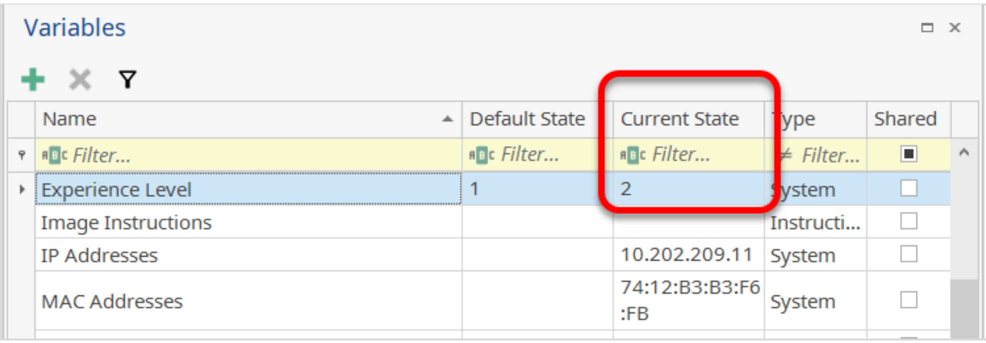
2. Under **Experience levels**, make sure the **Manual** radio button is selected.

3. Click the **Change current level...** button.



The **Variables** window opens. The variable **Experience Level** is selected.

4. Change the value in the column **Current State** to the experience level of your choice (1, 2 or 3).



The experience level has been changed.

Selection by the operator

When **manual** experience levels have been selected, you can choose to let the operator select his experience level himself.

To let the operator select the experience level, add experience level buttons to a **Screen** display.

For more information, refer to section **Adding experience level buttons** on page 138.

8. Variables

8.1 About variables

Variables are data items that can have different values or states during your assembly process.

Some variables are automatically generated by the Arkite software, for example detection and system variables. Other variables are created by the user.

The state of a variable can be used to log information or to trigger something in the process.

8.2 Variable types

In the table below, you can find a list of available variable types:

Detection type	Description
Detection variables (Validate and Vision license only)	Variables generated by the Arkite software for every detection that is added. – For every detection, there is a variable that indicates if the detection is present (ON) or not (OFF). – For every detection, there is an **occluder_** variable that indicates if the detection is occluded (ON) or not (OFF). – For every container, there is a **pick_** variable that indicates whether material is being picked from the container (ON) or not (OFF).
Instruction variables	Variables generated by the Arkite software for every process: – Text instructions variable, containing the currently displayed text instruction for the process – Image instructions variable, containing the currently displayed image instruction for the process
System variables	Variables generated by the Arkite software, for example: – IP addresses – MAC addresses – System Serial Number – Sensor Serial Number – Software Version – Current User – Current User Language – Current workstation – Project status – Current step [process] – Status [process] – Server Status – [Process] Step Start Time – [Process] Step StopTime – [Process] Step Duration – Experience Level – Active Variant – Projection Next Step – Projection After Next Step – Projection Error – Vis_[detection]: visualization for the detection projected or not – Sb_[detection]; shadowboard for the detection projected or not – [container]_material: material that is linked to the container

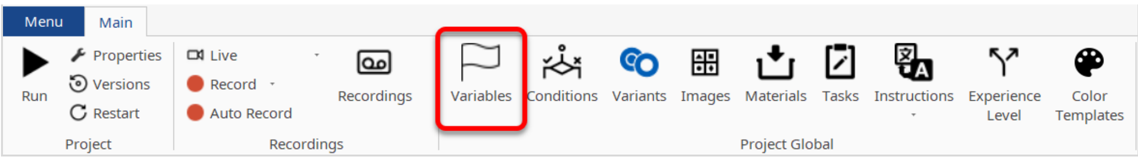
Detection type	Description
Communication variables	Variables that are generated by the Arkite software for communication with external devices through a communication module.
Condition variables	For each condition that is added by the user, a variable is generated. The value of this variable reflects the current status of the condition (**True** of **False**)
Manual variables	Variables that are added by the user, for example when configuring a communication module.

Note

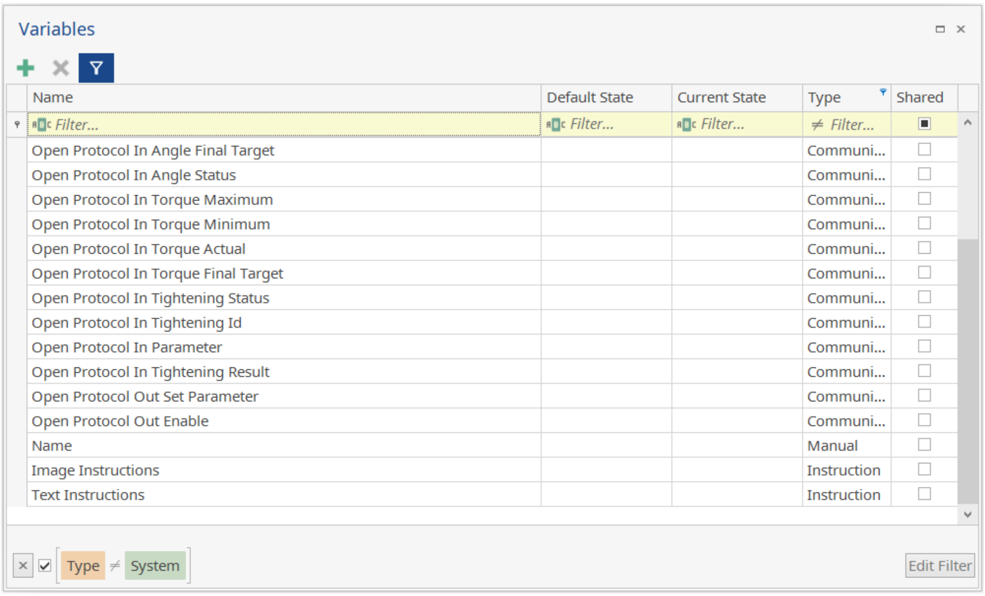
Only communication variables and manual variables can be updated by the user. All other variables can only be updated by the Arkite software.

8.3 Managing variables

In the ribbon, in the **Project Global** group, click the **Variables** button.



The **Variables** window appears.



For every variable, the information below is available

- **Name**
  For manual variables, the name can be changed.
- **Default state of the variable:**
  Initial state of the variable, when the project is started. Can be edited.
- **Current state of the variable:**
  Can be edited. When the project is restarted, the current state is always reset to the initial state.
- **Type:**

The variable type.

Note

*For an overview of variable types, refer to section **8.2 Variable types** on page 199.*

- **Shared:**

  If checked, the value of the variable is kept in sync across all workstations connected to the same server.

  If unchecked, the value of the variable can be changed independently for every workstation.

  Note

  *For more information, refer to section **Sharing variables** on page 53.*

By default, system variables are not shown in the overview.

Click to show system variables as well.

Click the **Edit Filter** button at the bottom of the window to define a more specific filter.

Click to add a new variable.

Click to delete a variable.

Note

Only manual variables can be deleted.

9. Recordings (Validate and Vision license only)

9.1 About recording

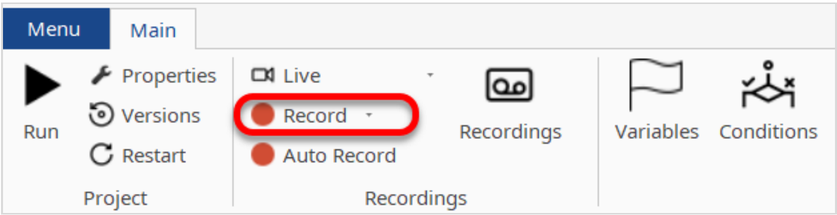
The Arkite software allows you to record the sensor stream. These recordings can be sent to Arkite for support, or you can use them to add detections from the comfort of your desk.

9.2 Creating recordings

You can create recordings in several ways:

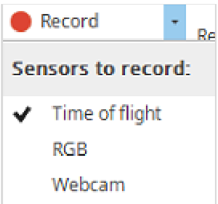
Record manually

1. In the ribbon, in the **Recordings** group, click the **Record** button.

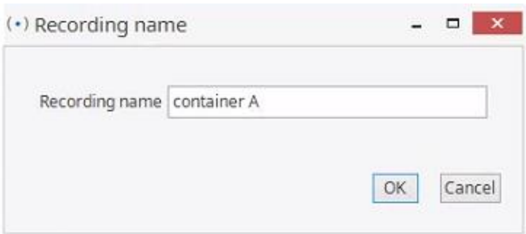


Note

To change the sensor(s) that is/are recorded, click the arrow next to the ***Record*** *button and select the sensor(s) of your choice.*



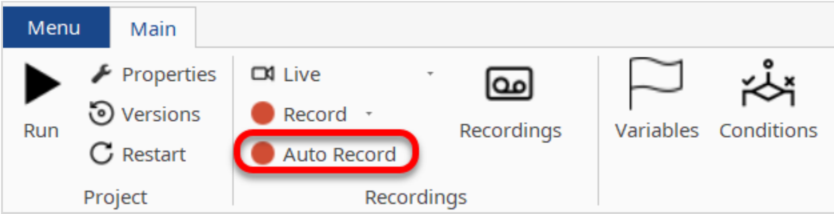
2. Perform the activities you want to record.
3. Click the **Record** button again to stop the recording.
4. Provide a name for your recording and click OK.



Your recording is saved.

Auto Record

1. In the ribbon, in the **Recordings** group, click the **Auto Record** button.



   A recording is made for every completed step involving a detection. Nothing is recorded during idle time.

2. Click the **Auto Record** button again to deactivate auto recording.

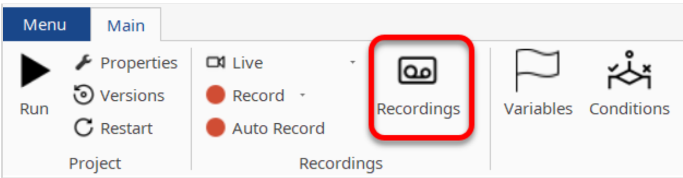
Use the Frames Recorder process step

Another way to trigger a recording is to introduce a **Frames Recorder** process step in your process at the moment you want the recording to start, and another one at the moment you want the recording to stop.

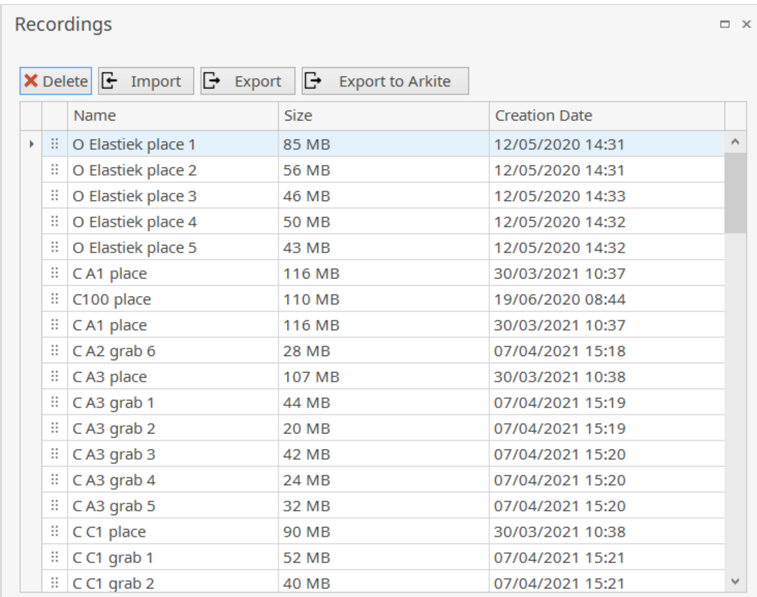
> ***Note***
> *For more information, refer to section* ***5.4.2 Adding process steps*** *on page 111.*

9.3 Managing recordings

To manage your recordings, click the **Recordings** button in the ribbon.



The **Recordings** window appears.



Name	Size	Creation Date
O Elastiek place 1	85 MB	12/05/2020 14:31
O Elastiek place 2	56 MB	12/05/2020 14:31
O Elastiek place 3	46 MB	12/05/2020 14:33
O Elastiek place 4	50 MB	12/05/2020 14:32
O Elastiek place 5	43 MB	12/05/2020 14:32
C A1 place	116 MB	30/03/2021 10:37
C100 place	110 MB	19/06/2020 08:44
C A1 place	116 MB	30/03/2021 10:37
C A2 grab 6	28 MB	07/04/2021 15:18
C A3 place	107 MB	30/03/2021 10:38
C A3 grab 1	44 MB	07/04/2021 15:19
C A3 grab 2	20 MB	07/04/2021 15:19
C A3 grab 3	42 MB	07/04/2021 15:20
C A3 grab 4	24 MB	07/04/2021 15:20
C A3 grab 5	32 MB	07/04/2021 15:20
C C1 place	90 MB	30/03/2021 10:38
C C1 grab 1	52 MB	07/04/2021 15:21
C C1 grab 2	40 MB	07/04/2021 15:21

To remove a recording, select it and click **Delete**.

To export a recording, select it and click **Export** or **Export to Arkite** to send to Arkite for support.

To import a recording from another project, click **Import** and select the .hrec file of your choice.

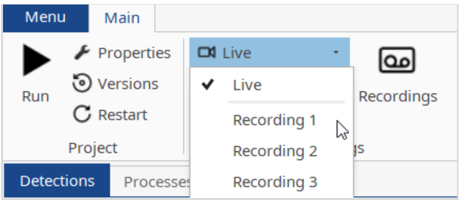
9.4 Using recordings

Recordings can be used by Arkite for troubleshooting and support.

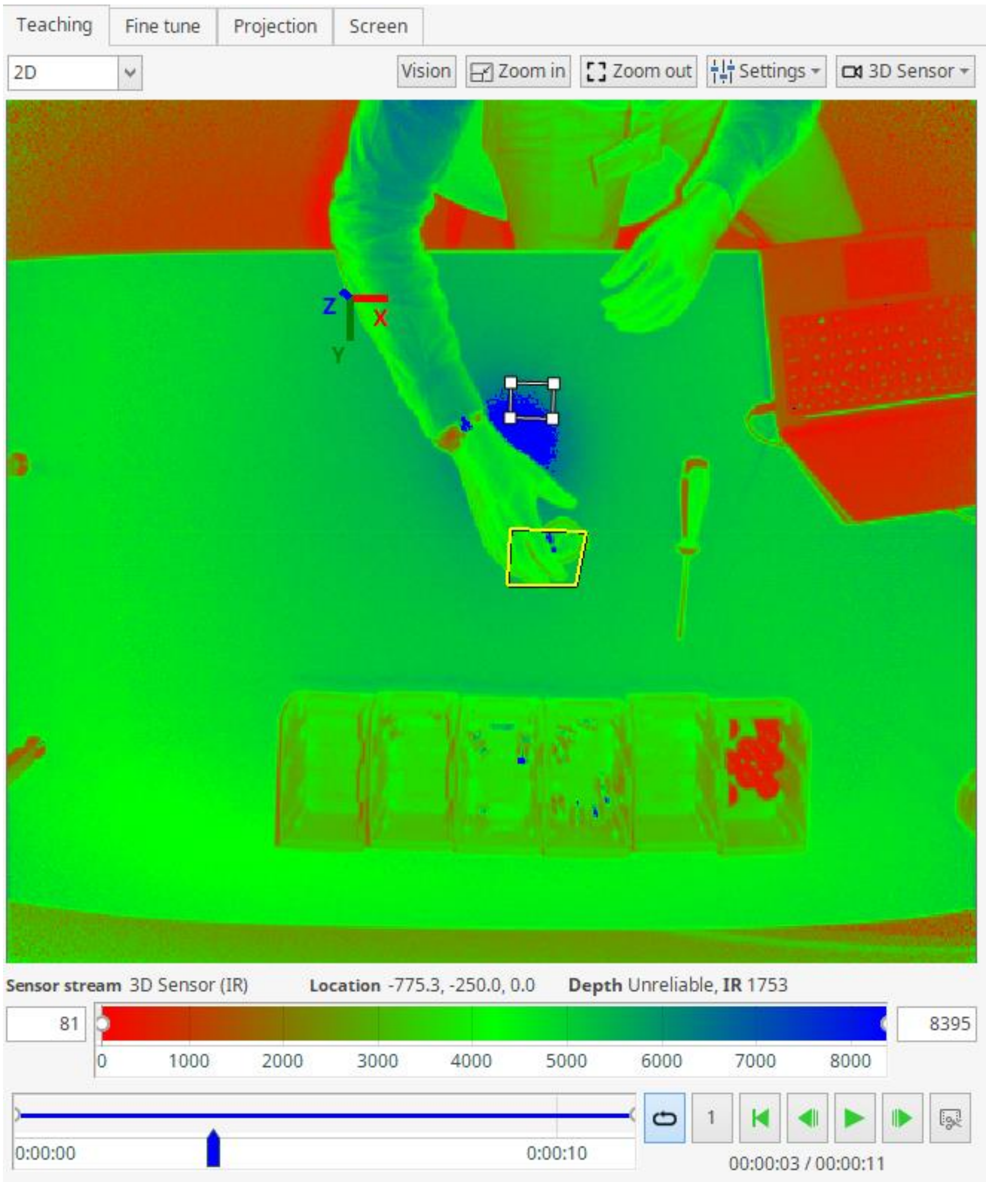
You can also use recordings yourself, for example for adding detections while you are not at the workbench.

To do this, execute the following steps:

1. At the workbench, make a recording of the sensor stream and make sure to remove all objects, tools and containers and put them back at the location of your choice one by one.
   Perform all activities for which you want to create a detection.
2. Open the project on your pc.
   Make sure you also import the recording.
3. In the ribbon, in the **Recordings** group, click the **Live** button and select your recording.



4. The recording starts to play in the **Teaching** subtab of the **Detections** tab.



5. To add detections, teach them by
   - pausing the recording at a frame where the object is not at its place and take a **Without object** snapshot
   - pausing the recording at a frame where the object is at its place and take a **With object** snapshot

10. Server management

10.1 About server management

The Arkite software is split in a workstation and a server component. The server, containing all data and resources for multiple clients and projects, should preferably run on a separate system to obtain higher security, maintainability and data integrity.

From within the software on any workstation connected to the server, you can consult an overview of all workstations on that server, manage licenses and manage backups.

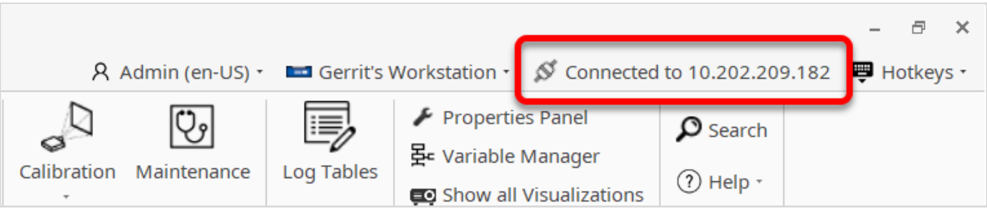
10.2 Connecting to a server

What?

When opening the Arkite software on a workstation the first time, you need to specify to which server you want to connect. Afterwards, the system automatically connects to that server again when starting the software. At all times however, you can connect to a different server.

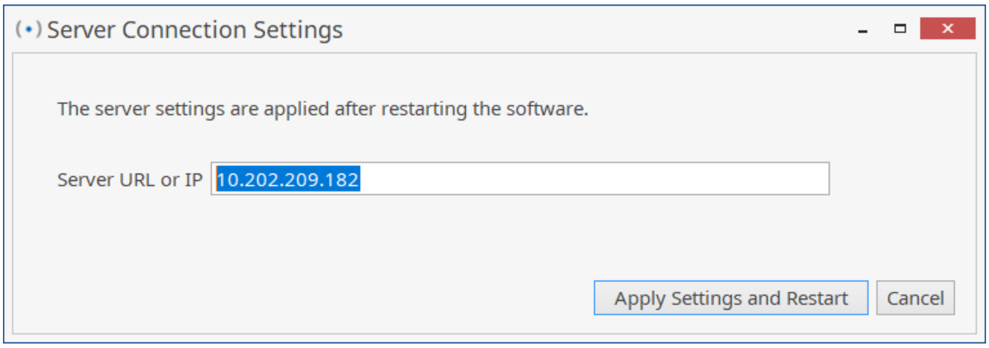
How to connect to a different server

1. In the top bar, click the **Connected to [server]** button.



The **Server Connection Settings** popup appears.

2. In the **Server URL or IP** field, fill in the URL or IP address of the server.



3. Click the **Apply Settings and Restart** button.

10.3 Workstation overview

What?

From any workstation connected to the server, you can consult information about all workstations that are connected to the same server, or have been connected to it in the past.

Overview of workstations connected to the server

To consult the overview of workstations, click the **Menu** tab and then **Workstations**.

Name	IP Address	Last Online Time	License	Software version	Logged in user	Up-time	Project loaded	Project loaded time
Filter...	Filter...	Filter...	Filter...	Filter...	Filter...	Filter...	Filter...	Filter...
1-3-2017034			With Sensor					
1-3-2018058	10.202.209.182	10/02/2022 16:36:16	With Sensor	2022.1.0.284	Admin	0:46:57	My assembly project	0:01:02
DESKTOP-FUS118A			None (editor only)					
Gerrit's Workstation	**10.202.209.16**	**10/02/2022 16:36:16**	**None (editor only)**	**2022.1.0.284**	**Admin**	**0:05:20**	**Project (2)**	**0:04:52**

In the overview table, you can see for every connected workstation:

- the workstation's **Name**.
  The name can be changed by clicking in the cell and entering a new name.
- the workstation's **IP address**
- the server **License** type that is used for the workstation:
  - Vision (= with Arkite 3D sensor and with Arkite Vision Sensor)
  - Validate (= with Arkite 3D sensor)
  - Guide (= without sensor)
  - none:
    in this case, the workstation is an editor. It can be used to prepare and configure projects, but not to run them. For testing purposes, you will be able to run your project for a maximum of 15 minutes.

Note

Workstations with an invalid license, for example a license that is only valid for a previous software version, are marked in red.

Name	IP Address	Last Online Time	License	Software version	Logged in user	Up-time
Filter...	Filter...	Filter...	Filter...	Filter...	Filter...	Filter...
LAPTOP BLA	**::1**	**30/05/2022 17:19:35**	**Invalid license**	**2022.2.9.9999**	**Admin**	**0:02:55**

- the **Software version** that is running on the workstation.
  A warning sign is displayed when the software version is not the same as the server software version.

Name	IP Address	Last Online Time	License	Software version	Logged in user
Filter...	Filter...	Filter...	Filter...	Filter...	Filter...
1-3-2017034			With Sensor		
1-3-2018058	10.202.209....	11/02/2022 10:59:56	With Sensor	2022.1.0.284	Standard User
DESKTOP-FUS118A			None (editor only)		
Gerrit's Workstation	**10.202.209.250**	**11/02/2022 11:43:42**	**None (editor only)**	**2022.1.0.286**	**Gerrit**

- The **user** hat is logged in on the workstation
- The workstation's **up-time**, i.e. the time it has been connected to the server
- The **name of the project** that is currently loaded on the workstation

- The **time** this project was loaded on the workstation
- **Vision Ready**: indicates if the workstation is ready to be used with an Arkite Vision license.
- **Host ID's**: ID that is required to make the workstation Vision Ready.

  Note

  *If you have a Vision license, but your workstations are not **Vision Ready** yet, follow the steps described in the procedure **How to make your workstation Vision Ready** (available on the customer drive). The number of workstations you can make **Vision Ready** is limited to the number of workstations for which you have bought a Vision license. If you already have reached this number, your request will be denied.*
- The **workstation id**

  You can copy this id to use it when working with the Arkite API. For more information about the Arkite API, refer to chapter **17 The Arkite API** on page 250.

The workstation from which the overview is consulted is displayed in **bold**.

Workstations that are currently not connected to the server are displayed in **grey**. These workstations can be turned off or no longer connected to the server. When hovering over a grey line, a tooltip is shown mentioning how long ago the workstation disconnected from the server. Workstations that are displayed in grey can be removed from the overview by selecting them and clicking the ✕ button.

10.4 Managing licenses

What?

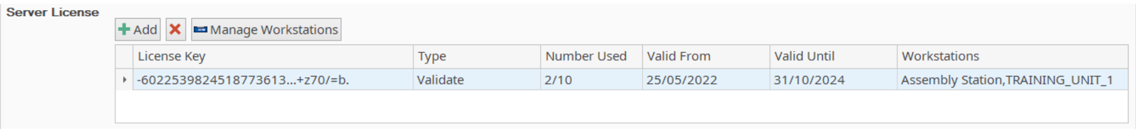
When using the workstation for the first time, you need to provide a server **License Key**. A license key is valid for a particular number of workstations. From the application, you can manage which workstations are using the license.

How to consult licenses

To consult licenses, click the **Menu** tab, then **Server** and then **Settings**.

Under **Server License**, you can see an overview of the license key(s) used for the server, their validity period, the number of workstations that can use the license and the workstations that are currently using the license.

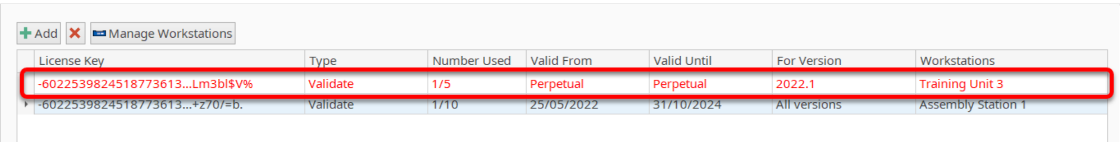
In the example below, one license key that can be used for a maximum of ten workstations is available on the server. Of those ten slots, two have already been assigned to the workstations with id's **Assembly Station** and **Training_Unit_1** respectively.



License Key	Type	Number Used	Valid From	Valid Until	Workstations
-6022539824518773613...+z70/=b.	Validate	2/10	25/05/2022	31/10/2024	Assembly Station,TRAINING_UNIT_1

Note

A license that is no longer valid is marked in red. In the example below, the license was only valid for a previous software version.



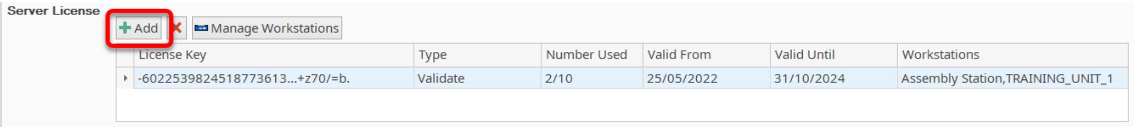
License Key	Type	Number Used	Valid From	Valid Until	For Version	Workstations
-6022539824518773613...Lm3bl$V%	Validate	1/5	Perpetual	Perpetual	2022.1	Training Unit 3
-6022539824518773613...+z70/=b.	Validate	1/10	25/05/2022	31/10/2024	All versions	Assembly Station 1

How to add a license key

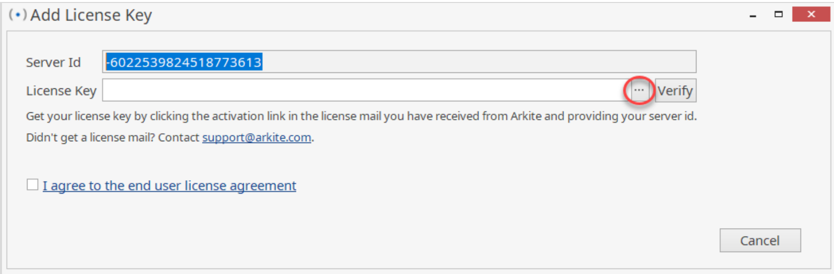
You can use several license keys on the same server.

To add a new **Server License Key**:

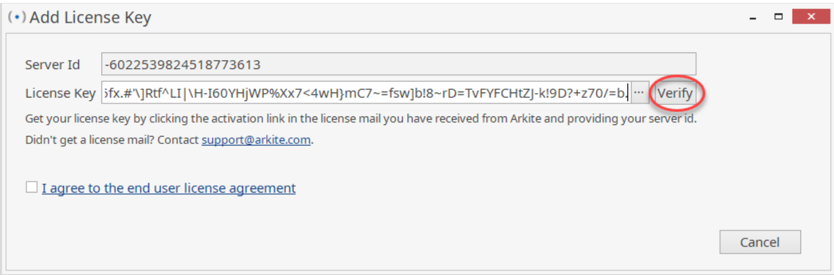
1. Click the **Add** button.



2. In the **Add License Key** window, in the **License Key** field fill in your license key or click the ... button to select the file containing your license key.

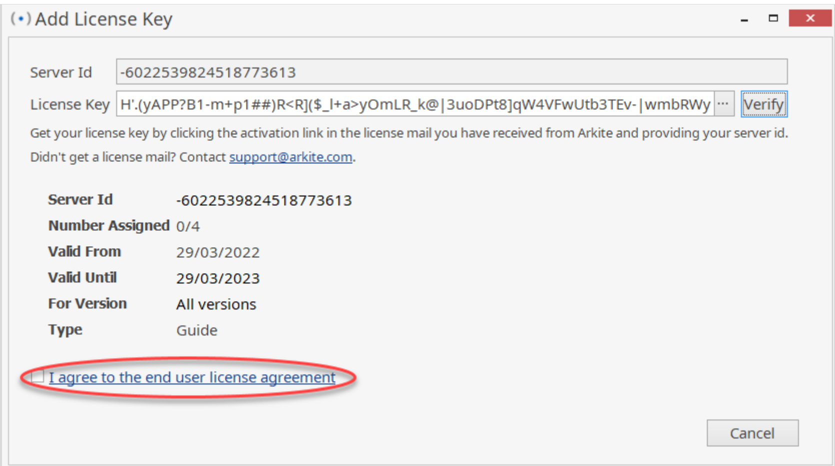


3. Click **Verify**.



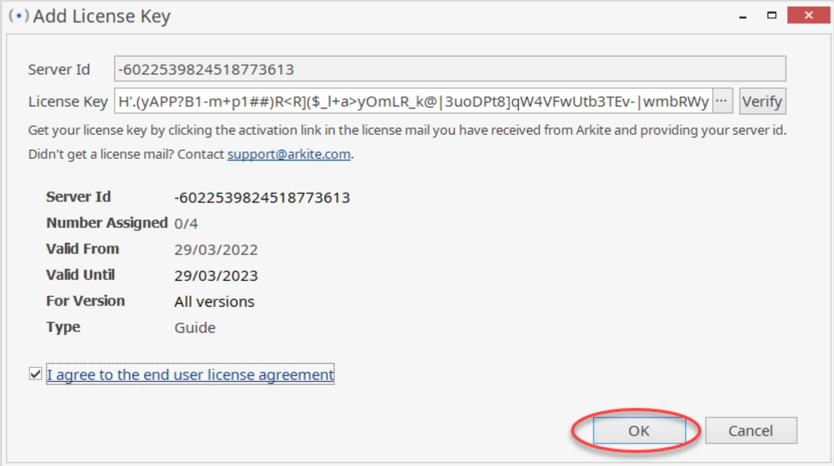
The details and validity date of your license is displayed.

4. Check the **I agree to the end user license agreement** checkbox.

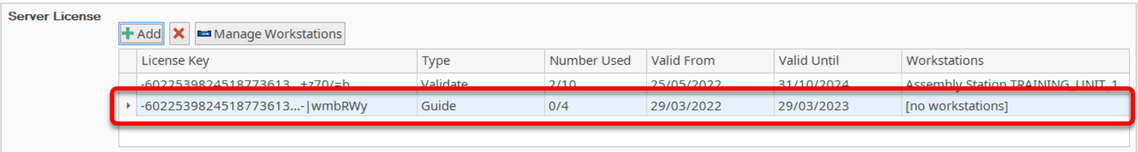


An **OK** button will appear.

5. Click **OK** to confirm.



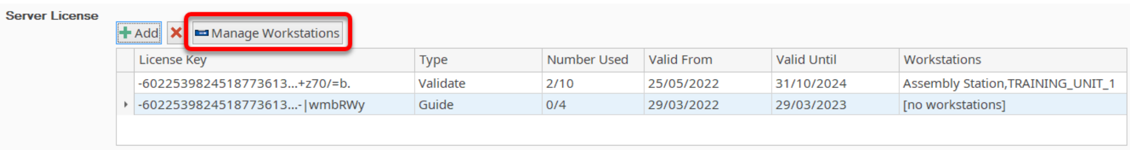
The license key is added to the overview.



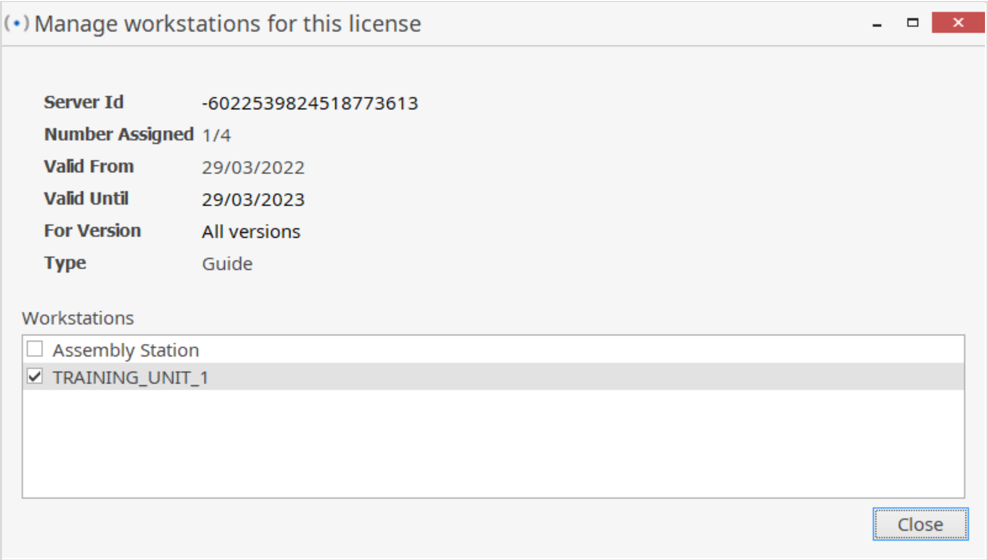
How to manage workstations

To manage the workstations on which the license key is used:

1. Select the license key and click the **Manage Workstations** button.



2. In the **Manage workstations for this license** window:
   - Check the checkbox(es) for the workstation(s) on which you want to use the license
   - Uncheck the checkbox(es) for the workstation(s) on which you no longer want to use the license



3. Click **Close** to apply your changes.

   Note

   - *If you have a Vision license, you will only be able to add workstations to it that are **Vision Ready**. If your workstations are not **Vision Ready** yet, follow the steps described in the procedure **How to make your workstation Vision Ready** (available on the customer drive). The number of workstations you can make **Vision Ready** is limited to the number of workstations for which you have bought a Vision license. If you already have reached this number, your request will be denied.*
   - *Workstations for which you have removed the license will lose the license after they have rebooted.*
   - *Workstations that are connected to the server but are not assigned to a license will function as an editor. You will be able to prepare and configure projects on them, but you cannot use them to run a project. For testing purposes, you will be able to run your project for a maximum of 15 minutes.*

How to renew a license key

Between three and two months before your license expires, you will receive a quotation from Arkite for the renewal of your contract. Once you have placed the purchase order, you will receive a new license key. You can then add this new license key as described above, without assigning any workstations to it yet.

Once your license key expires, the system will automatically check if another valid, comparable license key with open slots is available. If there is, it will automatically transfer the workstations to the new license key.

How to remove a license key

To remove a **Server License Key,** select the license and click the button.

10.5 Managing server backups

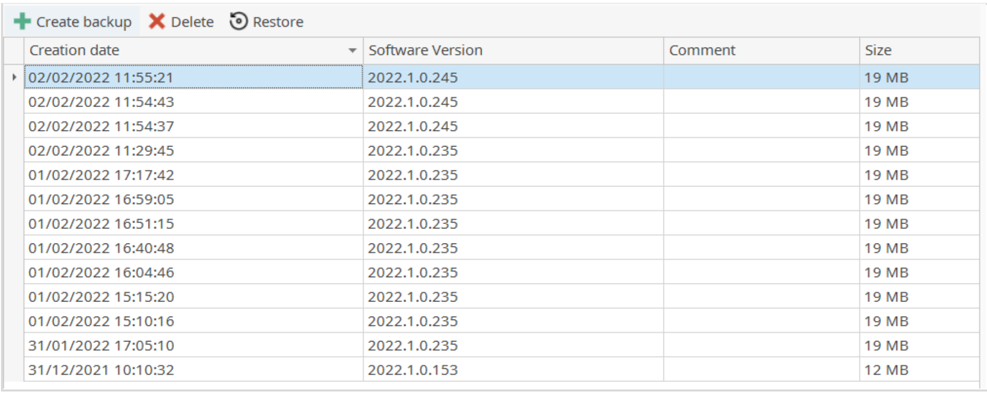
What?

Every five minutes, a backup of the server databases is taken (provided there have been any changes during this timeframe). The 10 most recent backups are always available. Additionally, all backups that were created right before a server version upgrade also remain available, as well as a selection of backups from the past. Other backups are removed automatically. You can consult all available backups and restore a previous version if required.

How to consult server backups

To consult server backups, click the **Menu** tab, then **Server** and then **Backups**.

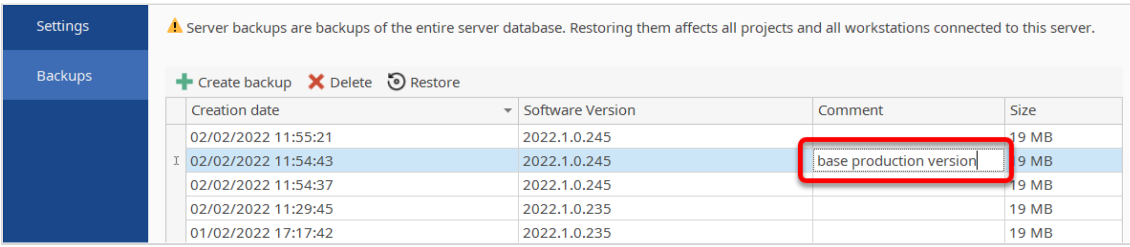
An overview of all available backups is shown.



Creation date	Software Version	Comment	Size
02/02/2022 11:55:21	2022.1.0.245		19 MB
02/02/2022 11:54:43	2022.1.0.245		19 MB
02/02/2022 11:54:37	2022.1.0.245		19 MB
02/02/2022 11:29:45	2022.1.0.235		19 MB
01/02/2022 17:17:42	2022.1.0.235		19 MB
01/02/2022 16:59:05	2022.1.0.235		19 MB
01/02/2022 16:51:15	2022.1.0.235		19 MB
01/02/2022 16:40:48	2022.1.0.235		19 MB
01/02/2022 16:04:46	2022.1.0.235		19 MB
01/02/2022 15:15:20	2022.1.0.235		19 MB
01/02/2022 15:10:16	2022.1.0.235		19 MB
31/01/2022 17:05:10	2022.1.0.235		19 MB
31/12/2021 10:10:32	2022.1.0.153		12 MB

For every backup, the information below is displayed:

- The backup's **creation date**
- The **software version** with which the backup was created
- **Comments** regarding the backup.
  You can add a comment for a backup by clicking in the corresponding cell in the Comments column and start typing.



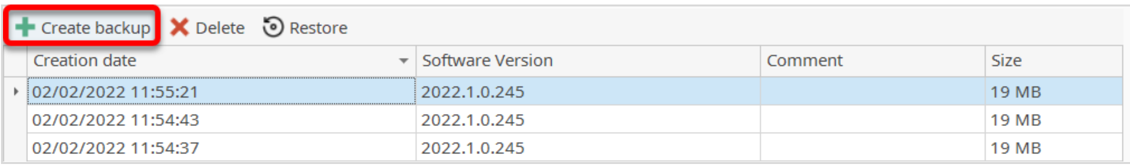
Creation date	Software Version	Comment	Size
02/02/2022 11:55:21	2022.1.0.245		19 MB
02/02/2022 11:54:43	2022.1.0.245	base production version	19 MB
02/02/2022 11:54:37	2022.1.0.245		19 MB
02/02/2022 11:29:45	2022.1.0.235		19 MB
01/02/2022 17:17:42	2022.1.0.235		19 MB

- The backup's **size**

How to create a server backup

While you are working on your project, the system automatically takes a backup every five minutes. Additionally, you can also create a backup yourself at any time.

To do so, click the **Create backup** button on top of the backup overview.



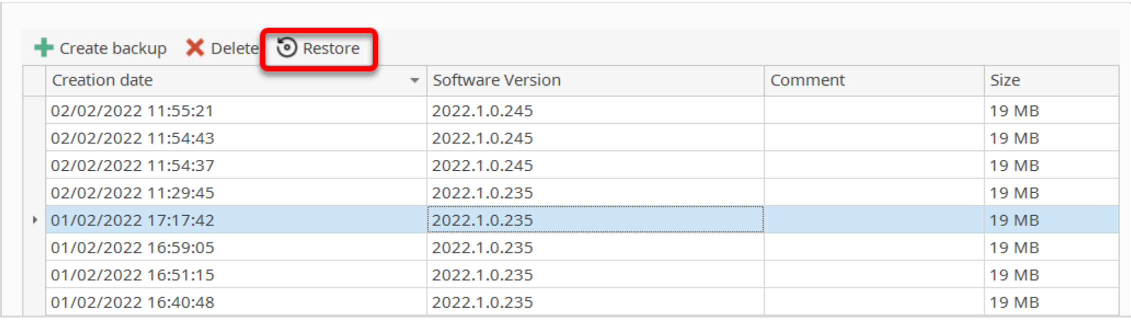
Create backup | Delete | Restore

Creation date	Software Version	Comment	Size
02/02/2022 11:55:21	2022.1.0.245		19 MB
02/02/2022 11:54:43	2022.1.0.245		19 MB
02/02/2022 11:54:37	2022.1.0.245		19 MB

How to restore a server backup

In case of issues, for example a corrupted database, you can put a previous version of your server database back in place by restoring a backup.

To do so, select the backup and click the **Restore** button.



Create backup | Delete | Restore

Creation date	Software Version	Comment	Size
02/02/2022 11:55:21	2022.1.0.245		19 MB
02/02/2022 11:54:43	2022.1.0.245		19 MB
02/02/2022 11:54:37	2022.1.0.245		19 MB
02/02/2022 11:29:45	2022.1.0.235		19 MB
01/02/2022 17:17:42	2022.1.0.235		19 MB
01/02/2022 16:59:05	2022.1.0.235		19 MB
01/02/2022 16:51:15	2022.1.0.235		19 MB
01/02/2022 16:40:48	2022.1.0.235		19 MB

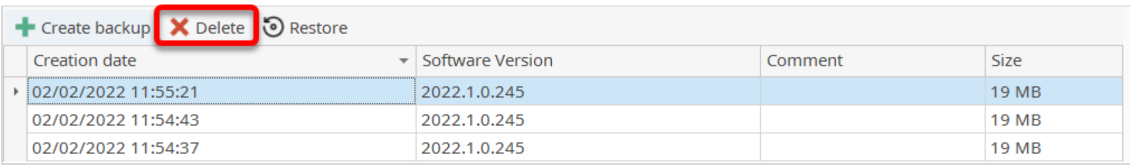
Warning

- *Server backups are backups of the entire server database. Restoring a server backup affects all versions of all projects on all workstations connected to the server. All workstations will need to restart their running project after the restore.*
- *Resources, such as image files, are not part of the server backups. This means that when you are restoring a server backup in which a resource file is used that has been removed from the project since, this resource file will no longer be available. An error message will be shown to inform you about the missing file(s).*

How to delete a server backup

Older backups are automatically deleted by the system. Additionally, you can also delete server backups yourself.

To do so, select the backup and click the **Delete** button.



Create backup | Delete | Restore

Creation date	Software Version	Comment	Size
02/02/2022 11:55:21	2022.1.0.245		19 MB
02/02/2022 11:54:43	2022.1.0.245		19 MB
02/02/2022 11:54:37	2022.1.0.245		19 MB

10.6 Cleaning up server resources

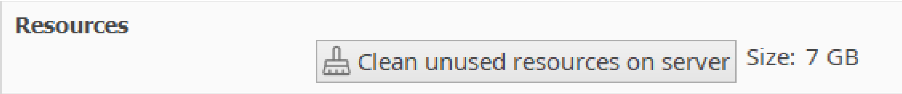
What?

When your disk is (almost) full, a clean up mechanism will remove all unused resources (e.g. image files). You an also trigger this mechanism yourself and free up disk space in this way.

Steps

1. Click the **Menu** tab and then **Server**.

2. Under **Resources**, click the **Clean unused resources on server** button.



11. Dashboarding

11.1 About dashboarding

Dashboarding provides you with insights on various aspects of your assembly process and the status of your system through a graphical representation of data.

11.2 Consulting dashboards

To consult dashboards, click the **Menu** tab and then **Dashboards**.

The dashboards below are currently available in the Arkite software:

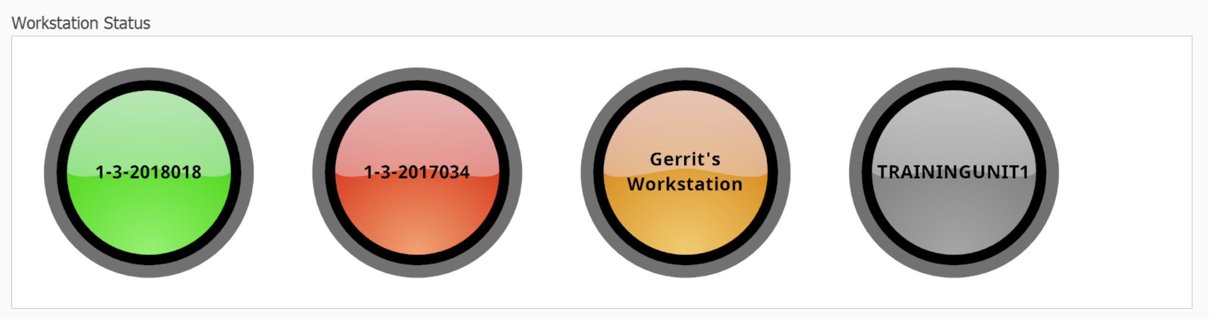
- **Workstation status:**
  provides you with information on the current status of each workstation that is connected to your server. Only workstations with a license are shown.
- **Product output:**
  provides you with information on the number of products that have been assembled during a specific period.
- **Cycle Times:**
  provides you with information on the time the assembly of a product has taken.
- **Current Step Duration:**
  provides you with real-time information about the current product assembly on the workstations connected to your server. Only workstations with a license are shown.

  Note

  A lot more information concerning product assembly is logged. You can use those to easily create your own dashboards in a dedicated tool.

11.3 Available dashboards

Workstation status



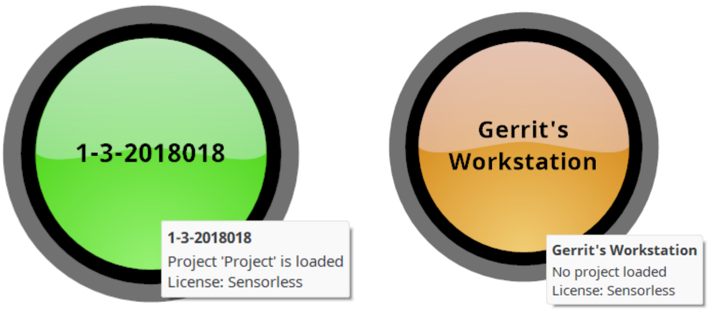
The workstation status dashboard provides you with information on the current status of each workstation connected to your server. Only workstations with a valid license are shown. In other words, editors are not displayed in the overview.

Each workstation is represented by a circle and can have four different colors:

- **Green:**
  the workstation is online, and a project is loaded.
- **Orange:**
  the workstation is online, but currently no project is loaded.
- **Red:**
  the workstation is online, but its software version does not match the software version on the server. It is not possible to load a project
- **Grey:**
  the workstation is currently not connected to the server.

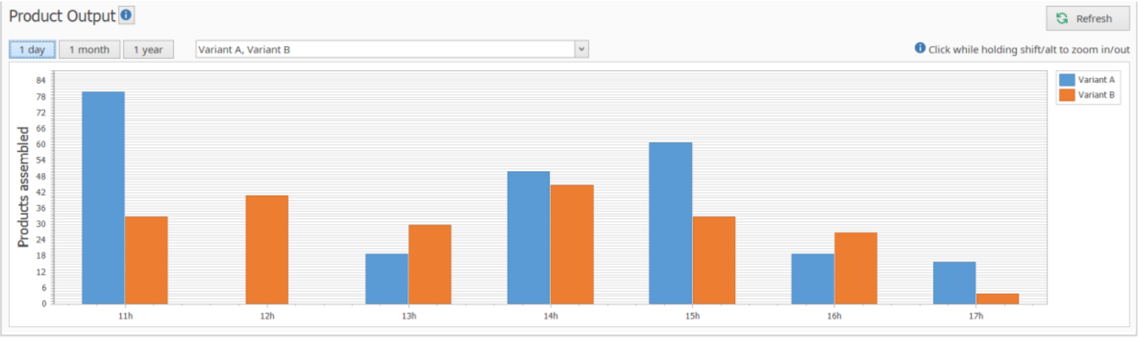
  Tip

  When you hover over a circle with your mouse, more information on the status and the workstation's license type is displayed in a tool tip.



Product Output

The **Product Output** dashboard provides you with information on the number of products that have been assembled during a specific period.



Click the **1 day** button to display the data for the last 24 hours. Data are split per hour.

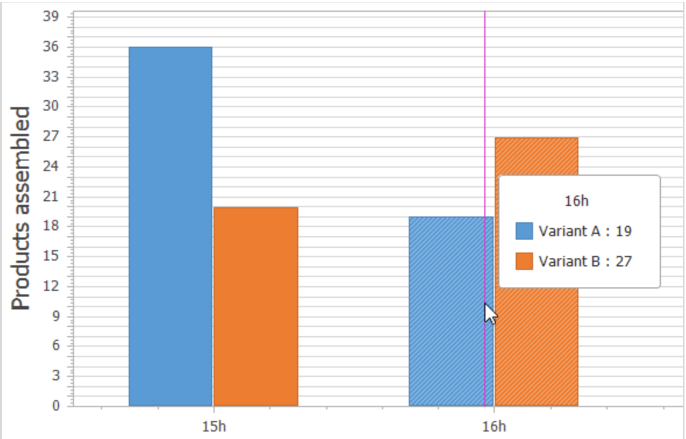
Click the **1 month** button to display the data for the last month. Data are split per day.

Click the **1 year** button to display the data for the last 12 months. Data are split per week.

In the **Products** dropdown list, you can select which products you want to see. By default, all products that have been assembled on a workstation connected to your server are shown.

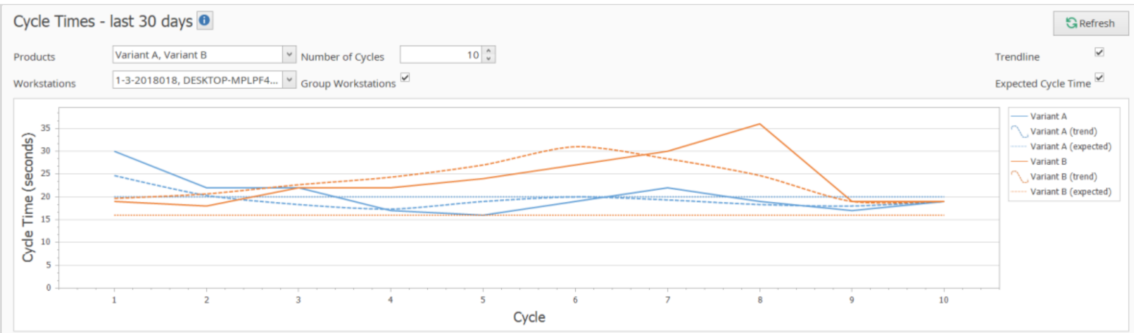
Note

- *This dashboard will only contain data if you have configured a **Log Product Start** and **Log Product Complete** process step in your assembly process. For more information, refer to section **5.4 Configuring process steps** on page 104.*
- *Click the **Refresh** button to update the dashboard with the latest data.*
- *When you hover over the graph with your mouse, the exact numbers for each bar in the graph are displayed in a tool tip.*



Cycle Times

The **Cycle Times** dashboard provides you with information on the time the assembly of a product has taken. By default, the last 10 assemblies of each product are shown. Only products that have been assembled during the last 30 days are taken into account.



In the **Products** dropdown list, you can select which products you want to see. By default, all products that have been assembled on a workstation connected to your server are shown.

In the **Workstations** dropdown list, you can select for which workstations you want to see the cycle times. By default, data for all workstations connected to your server are shown.

- When the **Group Workstations** checkbox is checked, one line for each product will be shown. You will see the last cycle times for the product, but you won't see on which workstation these assemblies have taken place.
- When the **Group Workstations** checkbox is not checked, a separate line for every workstation will be shown for each product. This allows you to compare cycle times between workstations for the same product.

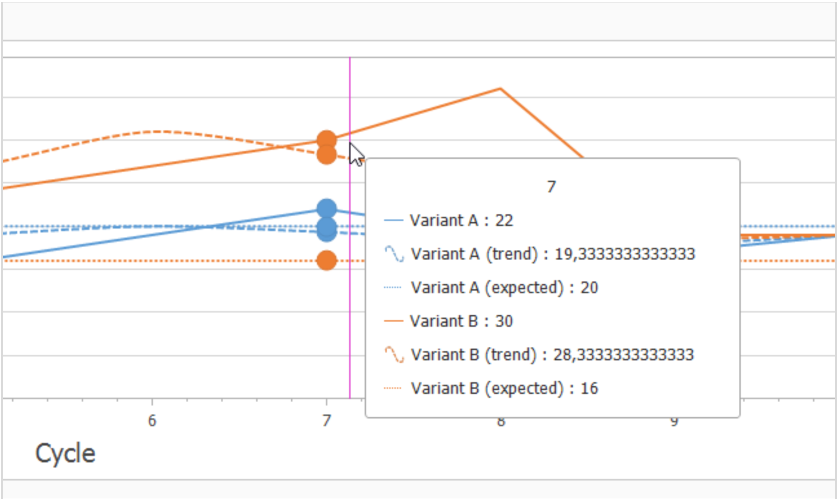
In the **Number of Cycles** dropdown list, you can select how many cycles are shown in the graph.

When the **Trendline** checkbox is checked, for each product or product/workstation combination a moving average is shown.

When the **Expected Cycle Time** checkbox is checked, for each product the expected cycle time is shown.

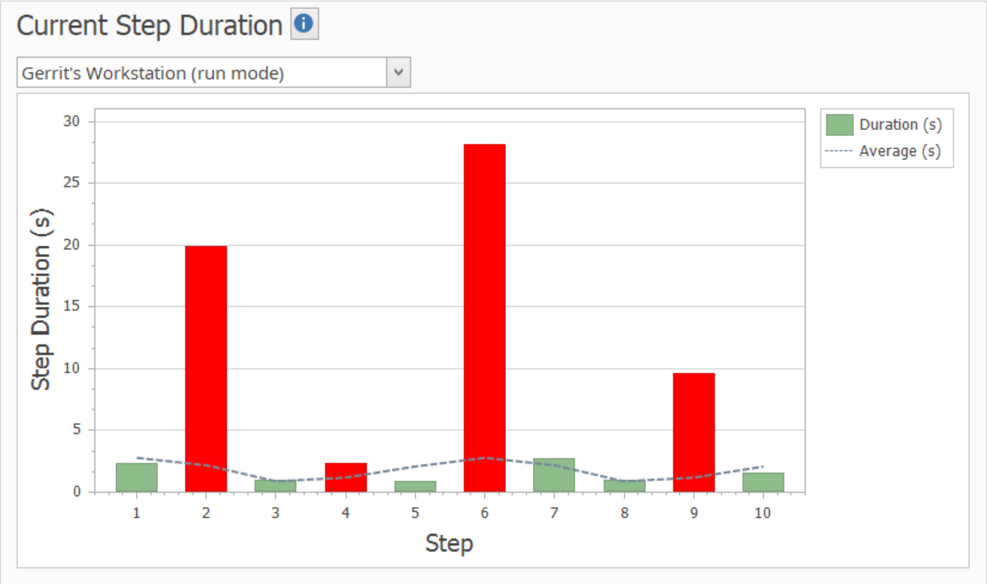
Note

- *This dashboard will only contain data if you have configured a **Log Product Start** and **Log Product Complete** process step in your assembly process. For more information, refer to section **5.4 Configuring process steps** on page 104.*
- *For more information on how to configure the expected cycle time for a product, refer to section **5.6 Working with variants** on page 139. It is not possible to configure an expected cycle time for products that are not based on the active variant in a project.*
- *Click the **Refresh** button to update the dashboard with the latest data.*
- *When you hover over the graph with your mouse, the exact numbers for each line in the graph are displayed in a tool tip.*



Current step duration

The **Current step duration** dashboard provides you with real-time information about the current product assembly on the workstations connected to your server. For each workstation with a license, the dashboard shows the duration of the step that is currently executed and the nine previous steps. Only steps with operator interaction are taken into account.



In the dropdown list above the dashboard, you can select the workstation for which you want to see the step duration.

For every step, you can see:

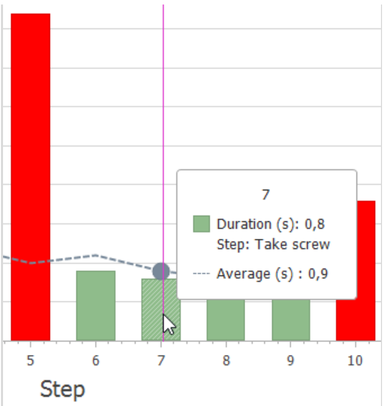
- The **average duration** of that step. The average is calculated based on the last 30 executions of that step. Outliers are not taken into account.
- The **duration in seconds** of the last time the step has been executed (or for the current step: the ongoing duration).

  In case the duration deviates significantly from the average, the bar for that step is shown in red.

  If the deviation from the averages is within range, the bar for that step is shown in green.

  Tip

  When you hover over a circle with your mouse, the step name and the exact duration and average duration are displayed in a tool tip.



12. Logging

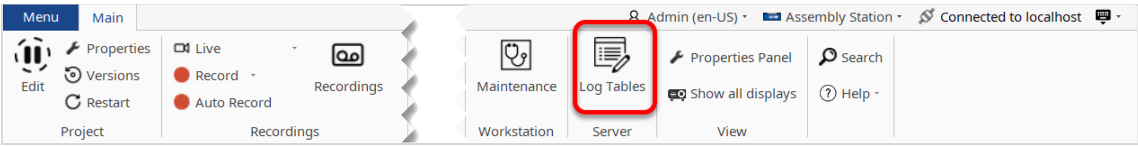
12.1 About logging

The server contains a database in which information about the platform and its projects are logged. Logging can be used to troubleshoot or reproduce issues, or to provide you with insights on various aspects of your assembly process and the status of your system through a graphical representation of data in dashboards.

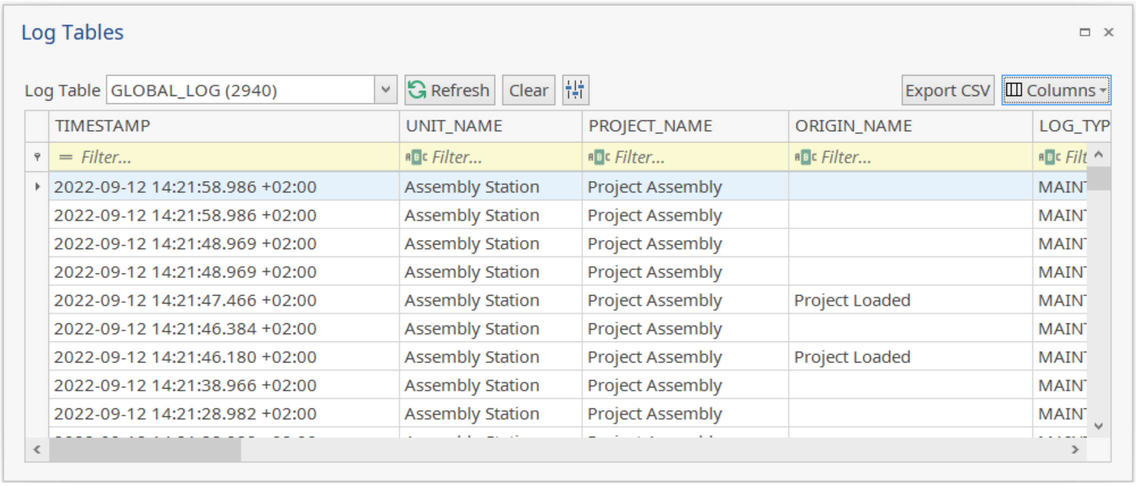
A lot of information is logged by default. Additionally, you can also add logging yourself

12.2 Consulting log tables in the Arkite software

To consult the log tables, click the **Log Tables** button in the ribbon.



The **Log Tables** window appears.



In the **Log Table** dropdown list, select the table you want to consult. By default, the log tables below are available:

Log table	**Description**
GLOBAL_LOG	Records are logged for error messages and default actions. ***Note*** *To change the default actions that are logged, refer to section* ***12.4 Changing logging settings for the GLOBAL_LOG table*** *on page 222.*
HEARTBEAT	Every 10 seconds, a record is logged when the workstation is online.
STEP_LOG	A record is logged for every step that is executed.
API_LOG	A record is logged for every communication with the Arkite API.

Log table	Description
PRODUCT_LOG	A record is logged every time a **Log product start**, **Log product complete** or **Log product scrap** is executed. The records in this table are used for generating the **Product Output** and **Cycle Time** dashboards. ***Note*** *For more information about dashboarding, refer to chapter **11 Dashboarding** on page 215.*
COMMUNICATION	Default log table for all communication modules in the project. A record is logged for every incoming and outgoing message, as well as error messages related to the communication module. Note For more information about communication modules, refer to section **6.3 Communication modules** on page 164.

Click **Refresh** to update the information in the log table.

Click **Clear** to remove the records from a log table.

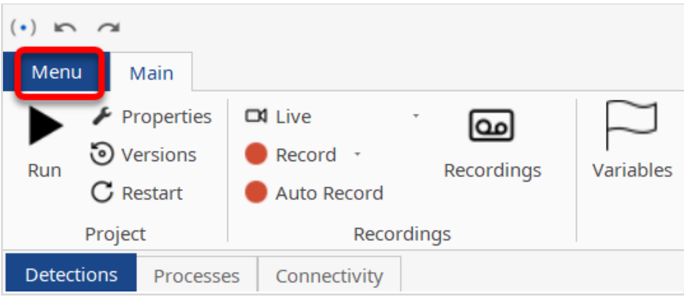
Warning

This action cannot be undone.

12.3 Changing the logging database and the logging database type

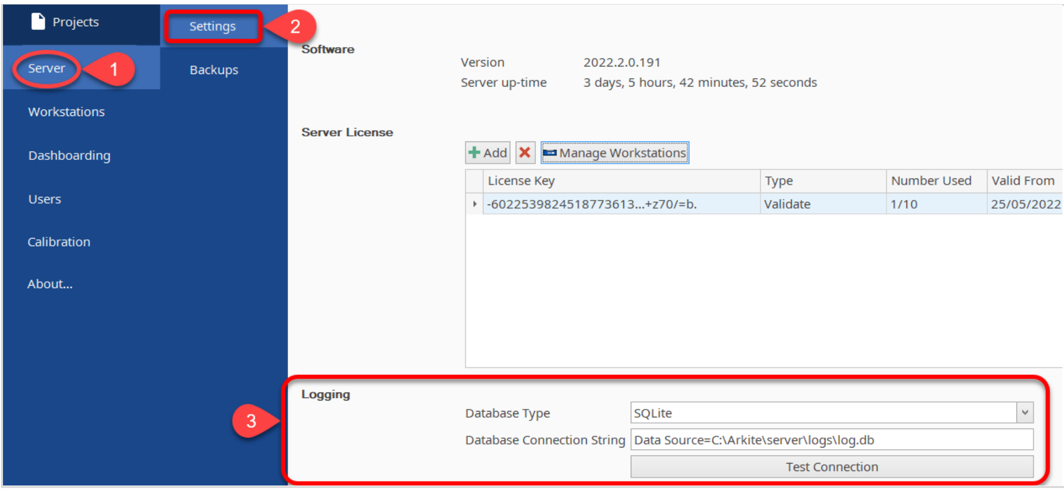
To change the logging database and/or the logging database type, execute the steps below:

1. Click the **Menu** tab.



2. Click **Server** and then **Settings**.

3. Under **Logging**, select the **Database Type** and enter the **Database Connection String**.



4. Click **Test Connection** to verify your configuration.

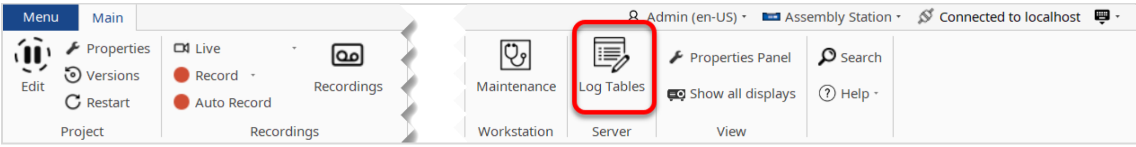
12.4 Changing logging settings for the GLOBAL_LOG table

What?

You can change the default actions that are logged in the **Global_LOG** table.

Steps

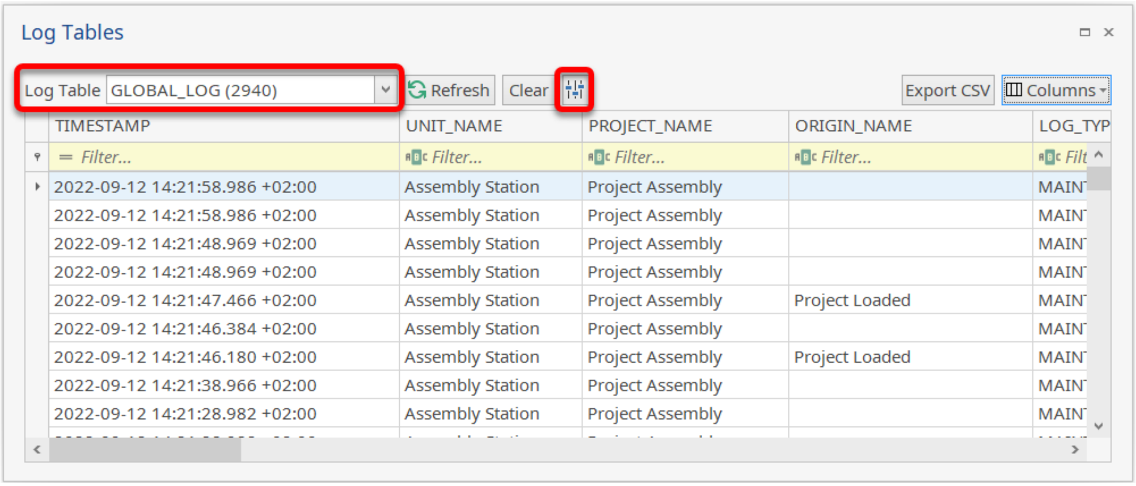
1. Click the **Log Tables** button in the ribbon.



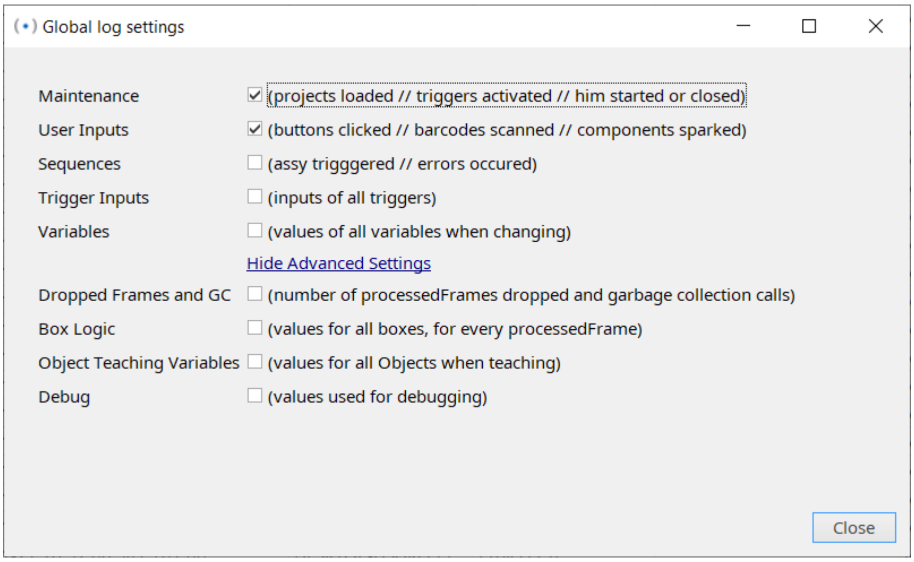
The **Log Tables** window appears.

2. In the **Log Table** dropdown list, select the **Global_LOG** table.

3. Click the button.



4. Check the checkboxes for the actions you want to log in the **Global_LOG** database.



5. Click **Close**.

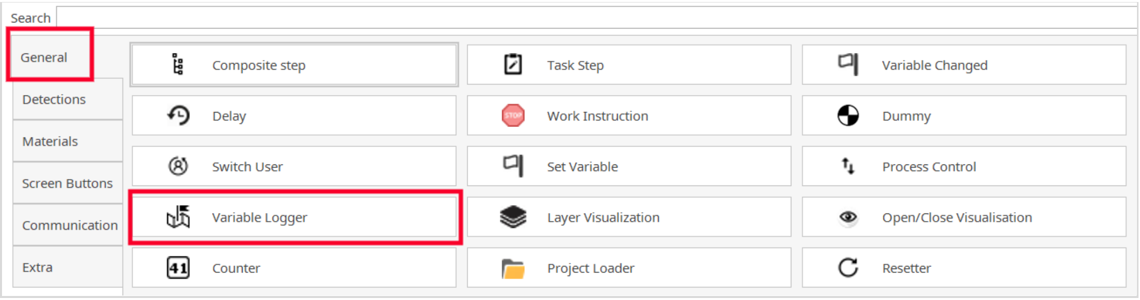
12.5 Adding logging yourself

Add a Variable Logger process step

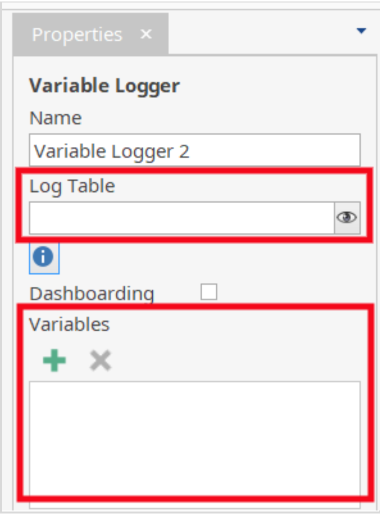
At any point in your process, you can add a step that logs the state of one or more variables:

1. Navigate to the **Processes** tab.
2. Under **Steps**, click the **New** button.

3. Select the **General** subtab and click **Variable Logger**.



4. In the **Properties** pane, add the variables for which you want to log the state and specify the **Log Table.**



Configure logging for a communication module

When configuring a communication module, you can specify a log table in which all incoming and outgoing messages and exceptions will be logged.

Note

Refer to section ***6.3.3 Adding a communication module*** *on page 178 for more information.*

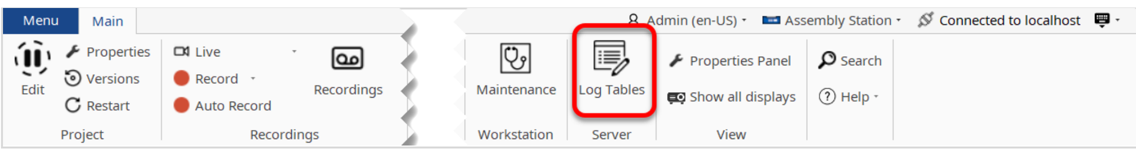
12.6 Exporting log data

What?

Log data can be exported to a csv file for further analysis in Microsoft Excel for example.

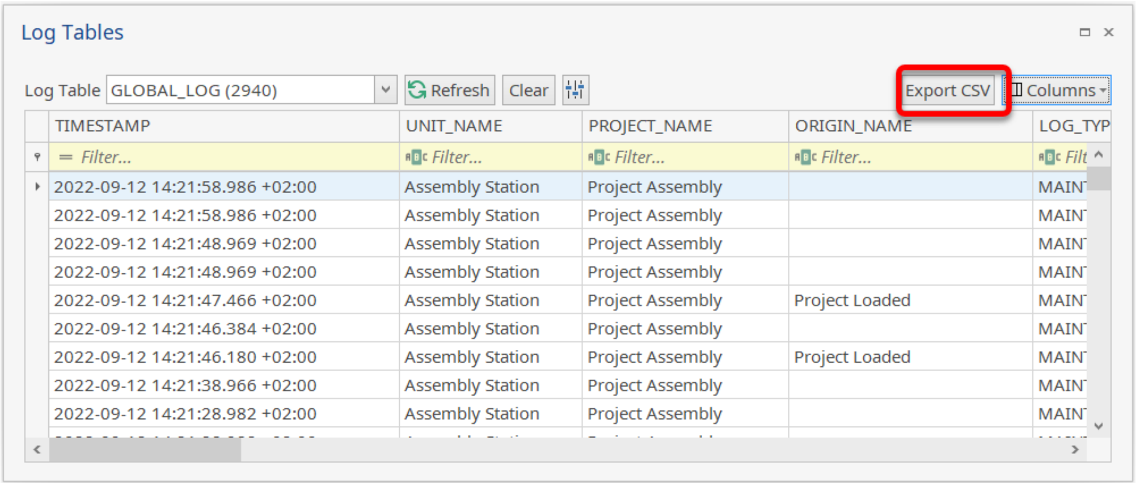
Steps

1. Click the **Log Tables** button in the ribbon.

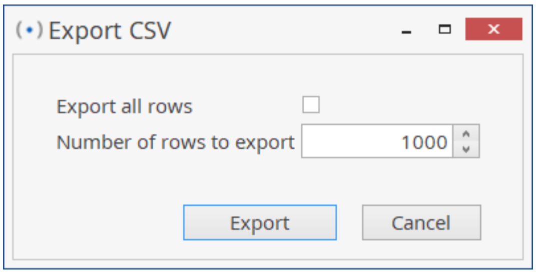


The **Log Tables** window appears.

2. In the **Log Table** dropdown list, select the table you want to export.
3. Click the **Export CSV** button.



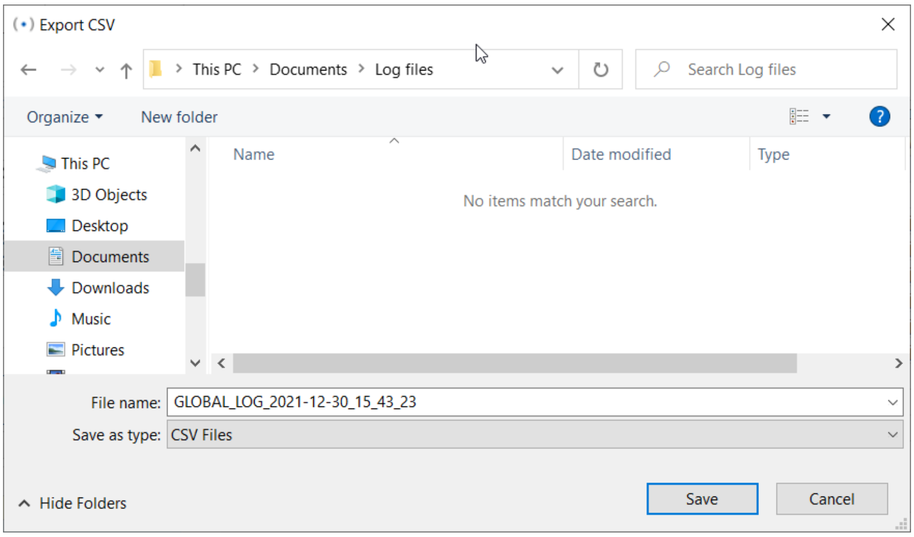
4. In the **Export CSV** window, indicate the number of rows you want to export or check the **Export all rows** checkbox to export everything.



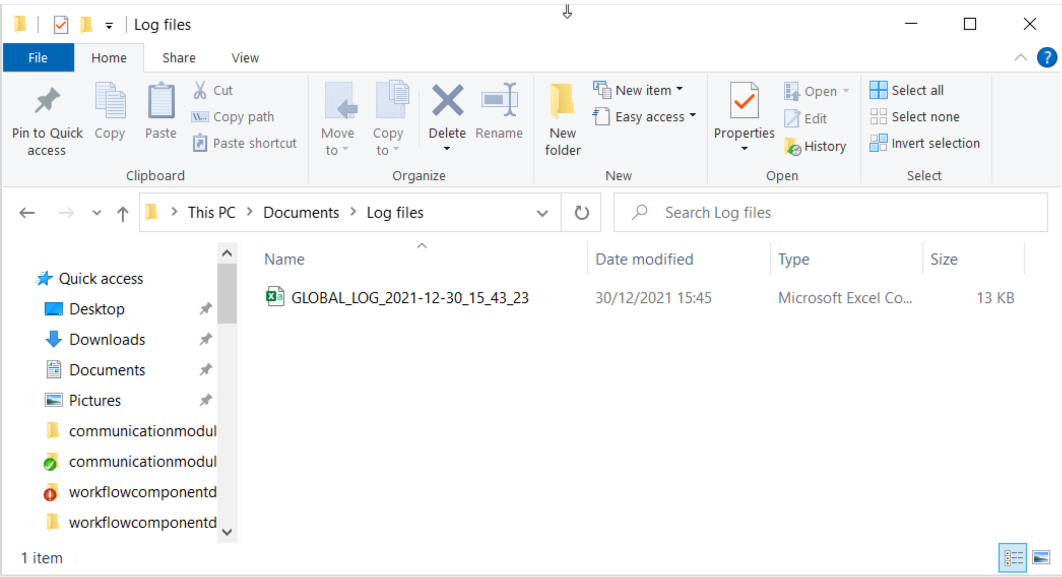
Note

If you select a number of rows, the most recent records in the database will always be exported.

5. Click **Export.**
6. In the **Export CSV** window, navigate to the folder of your choice and click **Save.**



The selected tables are exported to the directory of your choice.



13. Images

13.1 About images

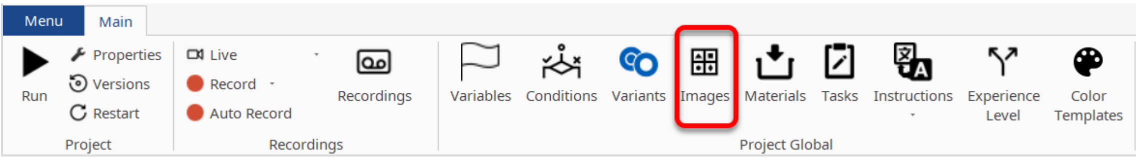
On several locations in the Arkite software, images and videos can be used, for example:

- Shadowboards for an object or tool
- Images linked to a detection
  These are projected on the workbench or shown on the screen when the detection is active.
- Image instructions linked to a process step
  These are projected on the workbench or shown on the screen when the process step is active.
- Images linked to a material
  These are displayed as an icon for containers containing this material
- ...

13.2 Managing images

Adding images or videos

1. In the ribbon, in the **Project Global** group, click the **Instructions** button.



The **Images** window appears.

2. Click the **New** button.

Several options are available:

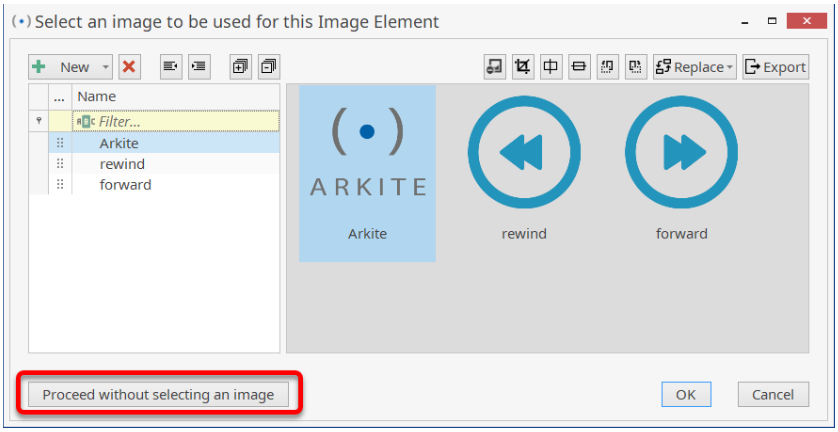
- **From File(s):**
  Allows you to add image files from your pc or a server for use in the Arkite software.
  Supported file types are: JPEG (.jpg, .jpeg), GIF (.gif), Bitmap (.bmp), PNG (.png), TIFF (.tiff, .tif), SVG (.svg), STA (.sta) and STL (.stl) – to be used for a CAD/CAM detection

For video files, .mp4 and .mov are supported

- **From Folder:**
  Allows you to add a folder from your pc. All image and video files from that folder will be added for use in the Arkite software.
- **From default images:**
  Allows you to add images from a library available to the Arkite software by default. It contains images that are commonly used, e.g. tools, icons, etc...
- **From RGB sensor– Validate and Vision license only:**
  Allows you to take a picture with the RGB sensor. You have the option to crop the image before you save it.
- **From RGB sensor (outline) – Validate and Vision license only:**
  Allows you to take a before and an after snapshot with the RGB sensor. An outline image will be created containing only the detected changes.

  Note

  *You can also use a variable to define which image or video to show in the **Image** visualization. To do so, click **Proceed without selecting an image**. In the properties of the **Image** visualization, you can define the variable to use. For more information, refer to section **5.5.1 Visualization types** on page 116.*



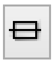
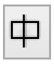
Organizing your images and videos

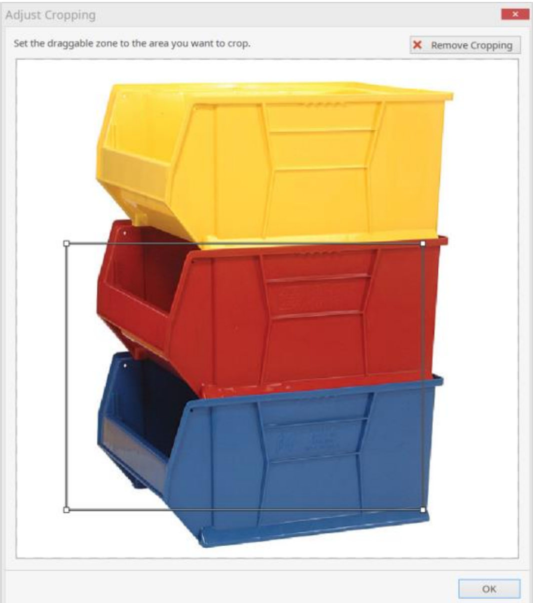
A number of buttons are available to help you organize your images and videos:

- Select an image or video and click ✕ to delete it.
- Select an image or video and drag it with your mouse to change its order.
- Select an image or video and click the button to create a subfolder and move the image or video into it.
  Once the subfolder exists, you can rename it and drag other images or videos into it.
- Select an image or video and click the button to move it up one folder.
- Click the button to increase the level of detail you see.
- Click the button to decrease the level of detail you see.

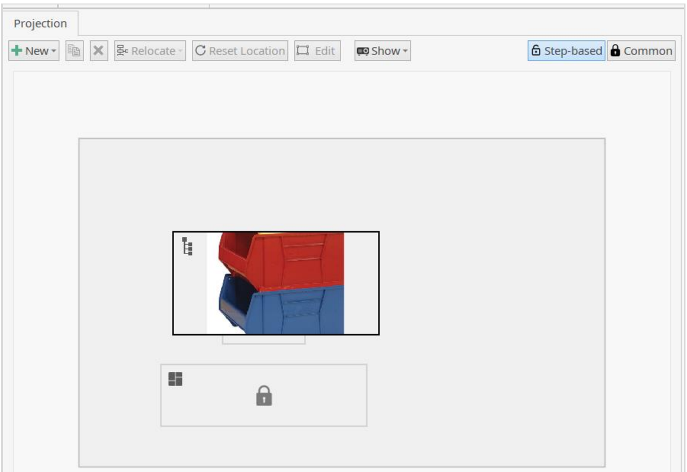
Editing images and videos

A number of buttons are available to edit images and videos you have added:

- Select an image and click  to mirror it horizontally or click  to mirror it vertically.
- Select an image and click  to rotate it clockwise or click  to rotate it counterclockwise.
- Select an image or video and click  to adjust the cropping.
  The **Adjust cropping** window appears. Set the zone you want to be visible and click **OK**.

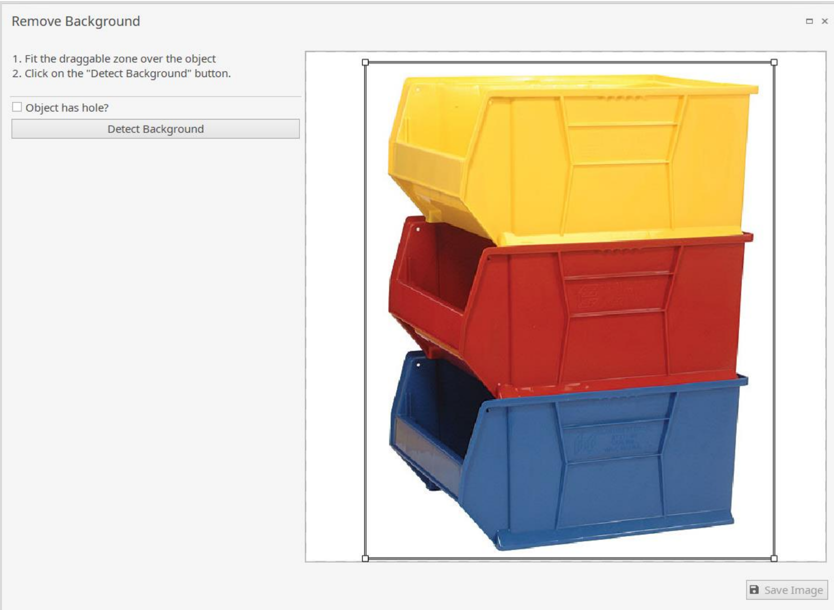


When you add the image, only the selected zone will be visible.

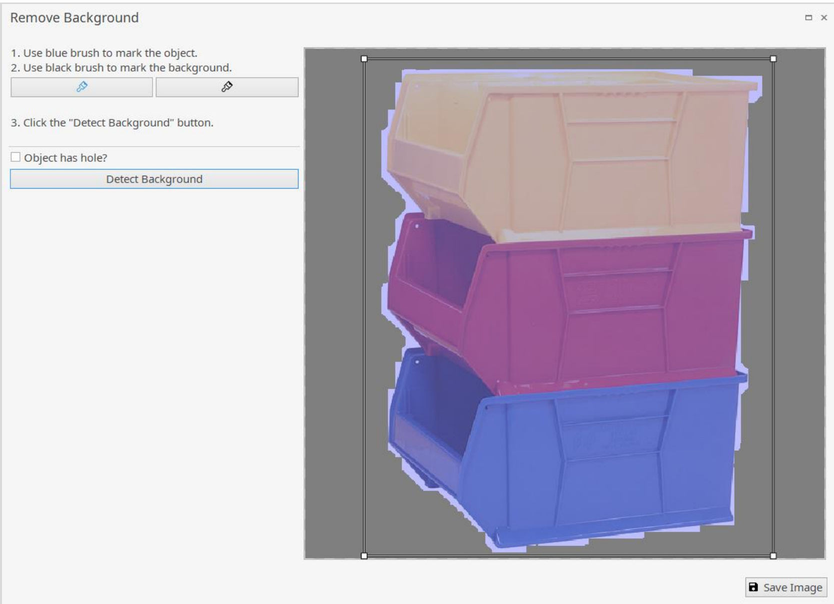


Later, you can still change or remove the cropping. That change will be applied on all locations where the image is used.

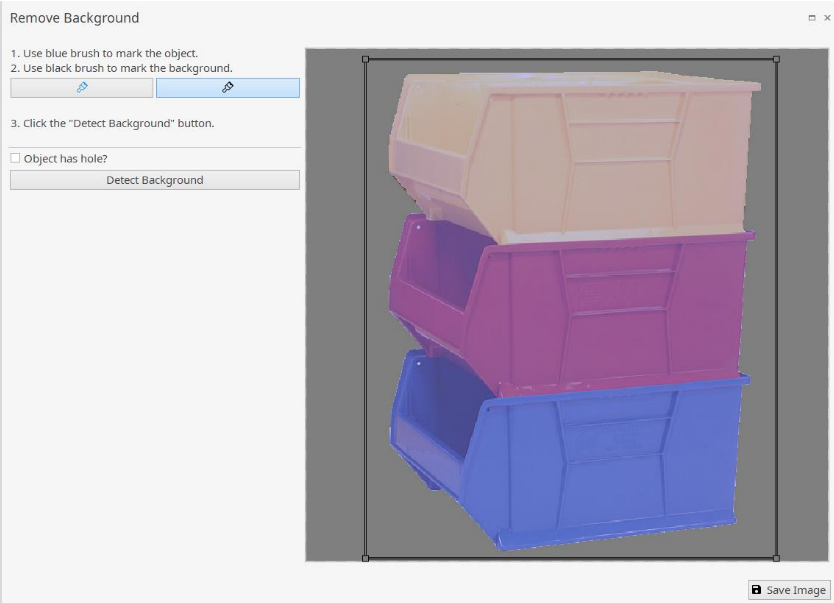
- Select an image and click  to remove its background.
  The **Remove background** window appears. Set the zone from which you want to remove the background and click **Detect background**.



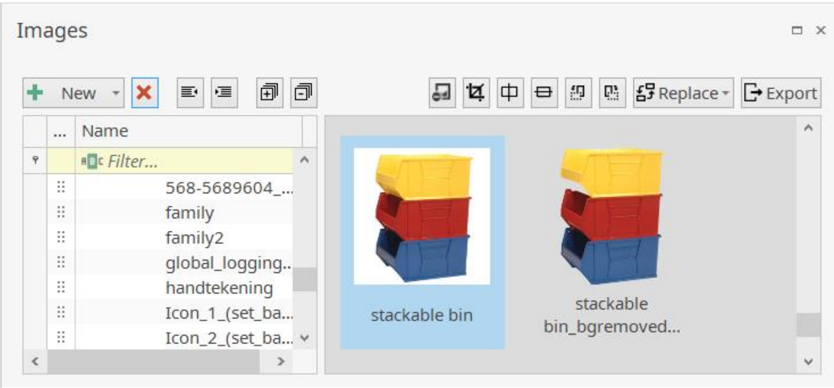
Afterwards you can finetune the background detection. Click the **black brush** button and indicate an area that is also part of the background but is not detected yet. Click the **blue brush** button and indicate an area that is part of the object but was mistakenly detected as background.



When you are satisfied with the result, click **Save Image**.



A copy of the image without background is saved with the extension **_bgremoved** added to the filename.



Exporting images and videos

Select an image or video and click **Export** to export it.

Your image or video will be exported to the folder of your choice.

14. Color templates

14.1 About color templates

Color templates can be used to define a visualization's color based on the value of a particular variable. When changing or updating the template, all visualizations linked to the template will be updated automatically.

Note

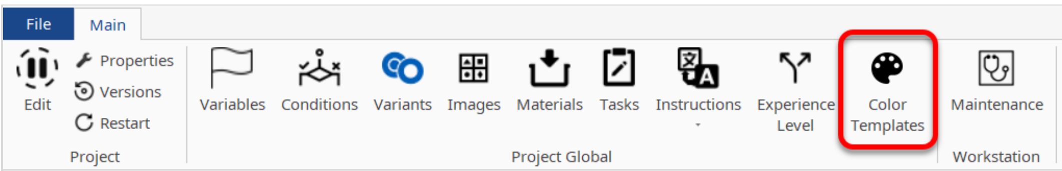
*You can also define a visualization's color based on a variable without using templates. In that case, changes need to be applied for every visualization separately. For more information, refer to section **Defining the color based on the value of a variable** on page 133.*

14.2 The Detection Colors template

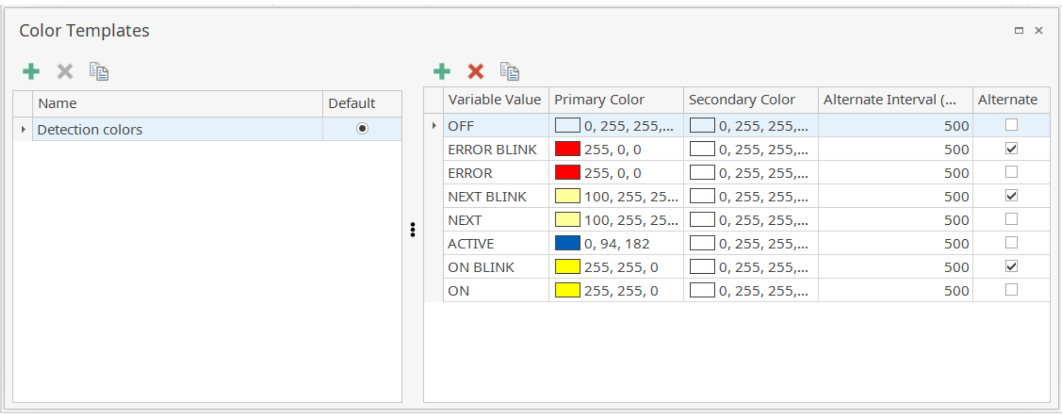
By default, one color template called **Detection colors** is available in the Arkite software. This template is used to define the colors of visualizations on a **Projector** display linked to a detection during the execution of your assembly process.

To consult the Detection Colors template:

1. In the ribbon, in the **Project Global** group, click the **Color Templates** button.



2. In the **Color Templates** window, in the left pane, select **Detection Colors**.
3. In the right pane, you can see the colors as defined for each status of the detection.



Note

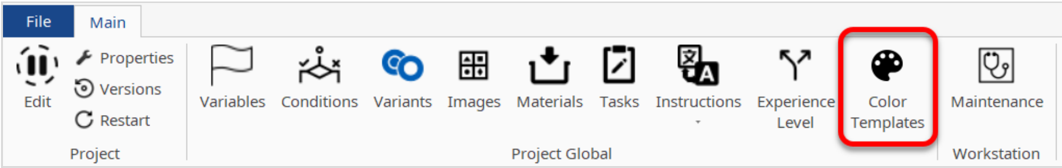
- *In case the **Alternate** checkbox is checked, the color of the projection will alternate between the **Primary Color** and the **Secondary Color**. By default, the **Secondary Color** is transparent.*
- *In case the **Alternate** checkbox is not checked, the **Primary Color** will always be used.*

The detection's status is defined by the active step in the assembly, combined with the projection settings. To learn more about the projection settings, refer to section **5.5.7 Managing projection settings** on page 136.

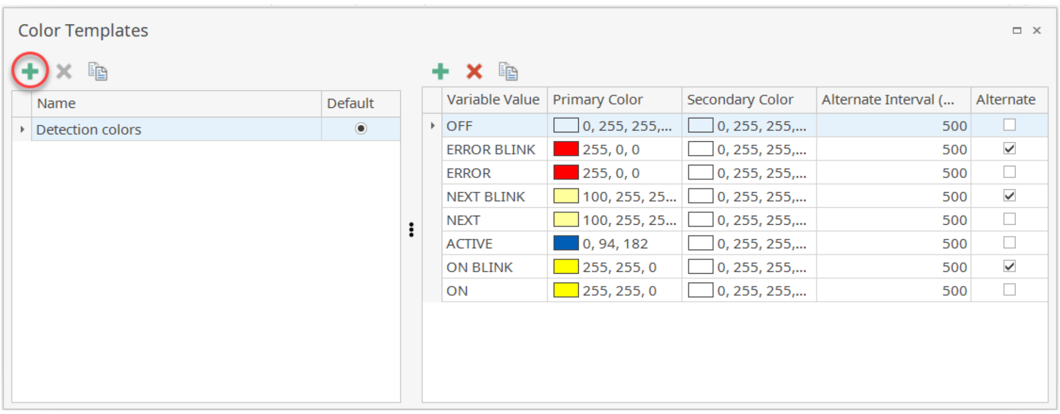
14.3 Managing color templates

Adding color templates

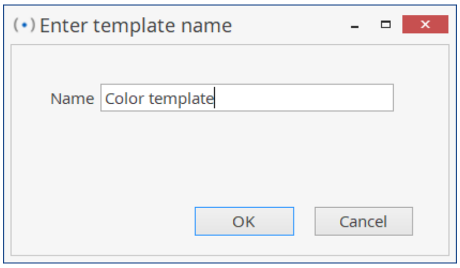
1. In the ribbon, in the **Project Global** group, click the **Color Templates** button.



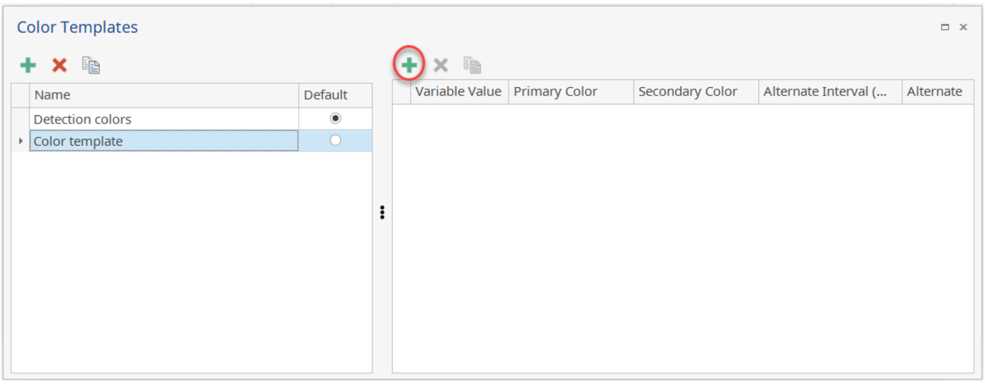
2. In the **Color Templates** window, click the ➕ button in the left pane.



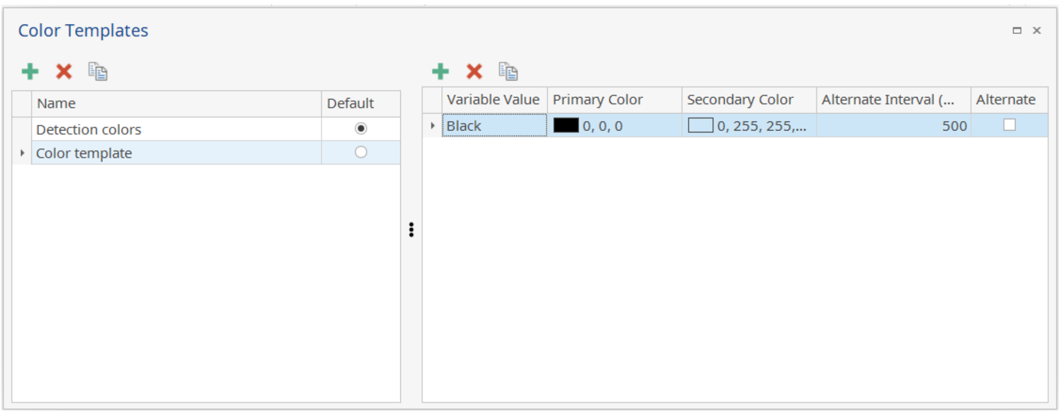
3. In the **Enter template name** window, enter a name for your template and click OK.



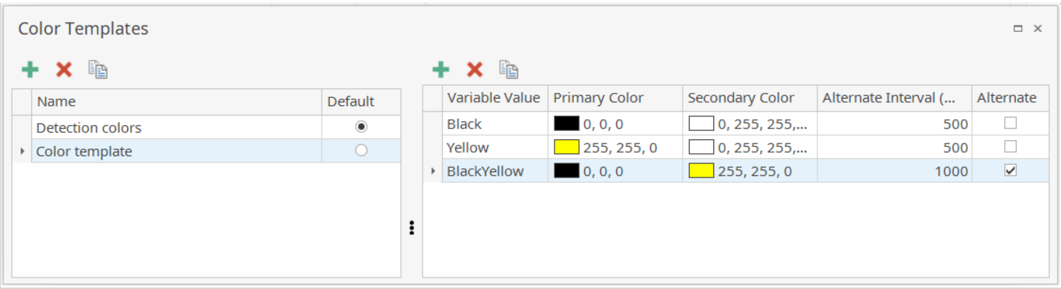
4. Click the ➕ button in the right pane.



5. Enter the information below:
   - In the **Variable Value** column, enter a value for which you want to define a color.
   - In the **Primary Color** column, specify the color you want to use.
   - If you want to alternate between two colors, check the **Alternate** checkbox, specify another color in the **Secondary Color** column and select an **Alternate Interval** (250, 500 or 1000 milliseconds). If you don't check the **Alternate** checkbox, the values in the **Secondary Color** and **Alternate Interval** columns are ignored.



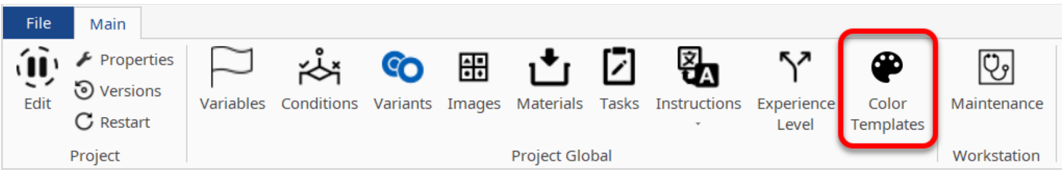
6. Repeat step 5 for all required values.
   In the example below, the visualization's color will be yellow when the value of the **Color** variable is *Yellow* and black when the value of the **Color** variable is *Black.* When the value of the **Color** variable is *BlackYellow*, the color will alternate between black and yellow every second.



Note
*For each color property where you want to use this template, you will have to specify a variable that contains the values defined in the template. For more information, refer to section **Defining the color based on a template** on page 134.*

Editing color templates

1. In the ribbon, in the **Project Global** group, click the **Color Templates** button.



2. In the **Color Templates** window, in the left pane, select the template you want to edit.
3. In the right pane, add, remove or modify the variable values of your choice.

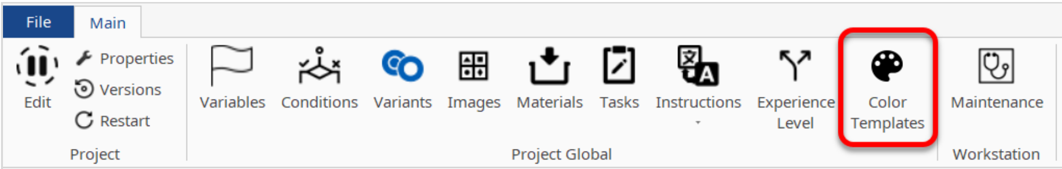
Setting a color template as default

By default, the **Detection colors** template is used to define the colors of visualizations on a **Projector** display linked to a detection during the execution of your assembly process.

You can configure another template to use for this purpose. It is important that all values used by the projection settings are available in this template. Therefore, it is advised to duplicate the **Detection Colors** template and make the required changes in there.

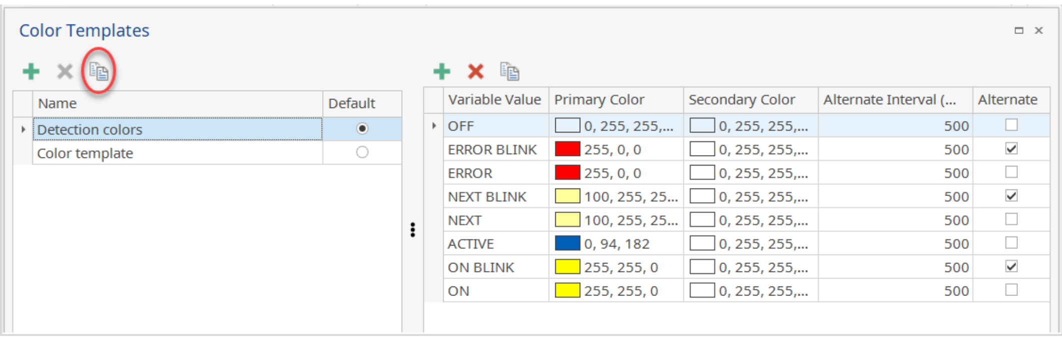
Execute the steps below:

1. In the ribbon, in the **Project Global** group, click the **Color Templates** button.

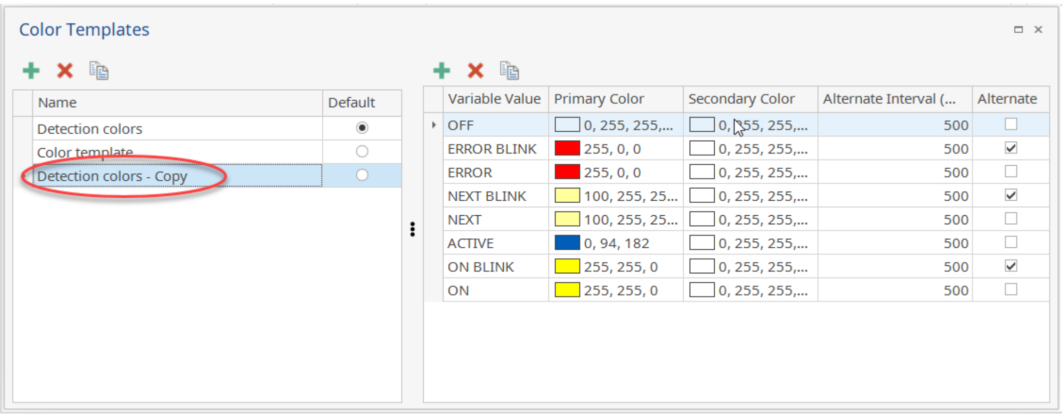


2. In the **Color Templates** window, in the left pane, select **Detection Colors**.

3. Click the  button in the left pane.



4. In the left pane, rename the **Detection Colors – Copy** template if required.

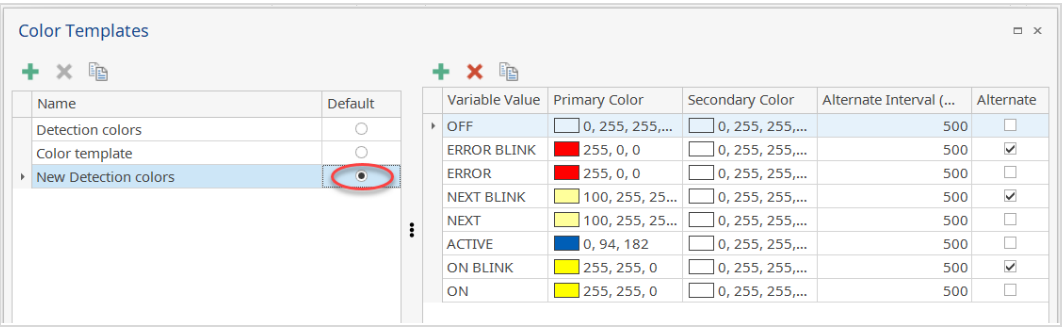


5. In the right pane, modify the colors for the required variable values.

   Important

   For the detection to work properly, do not remove any of the variable values from this template.

6. In the left pane, select the radio button in the **Default** column for the new template.



The new template will now be used for visualizations on a **Projector** display linked to new detections. Visualizations linked to existing detections will still use the original **Detection colors** template. If you want to also use the new template for these visualizations, update their **Brush Color** property, as explained in section **Defining the color based on a template** on page 134.

15. Languages

15.1 About languages

The Arkite software allows you to define text instructions in different languages. This way, every operator can be supported in his or her own language while executing a process.

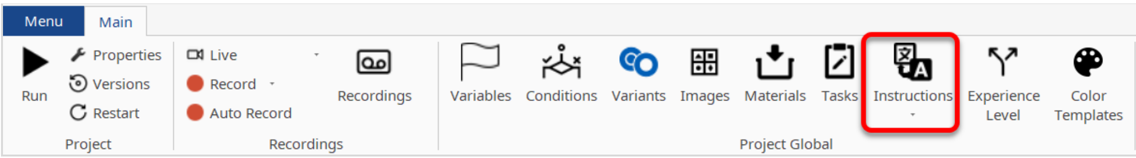
By default, the Arkite software allows you to add instructions in English (en-us), but you can add as many languages as you want.

> ***Note***
>
> *By default, instructions will be displayed in the language of the user that is running the project. If no instructions are available in that language, the system will look for a similar language. For example, if the user language is en-GB, en-US will be used. If no similar language is available, the first instruction language in the project will be used. For more information on defining the user language, refer to section **16.4 Managing local users** on page 246.*

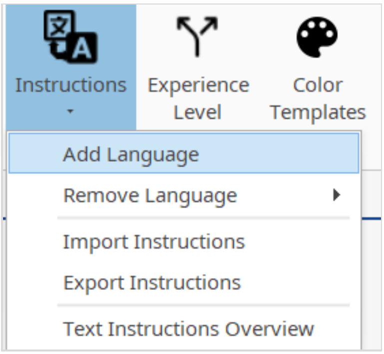
15.2 Managing languages

Adding a language

1. In the ribbon, in the **Project Global** group, click the **Instructions** button.

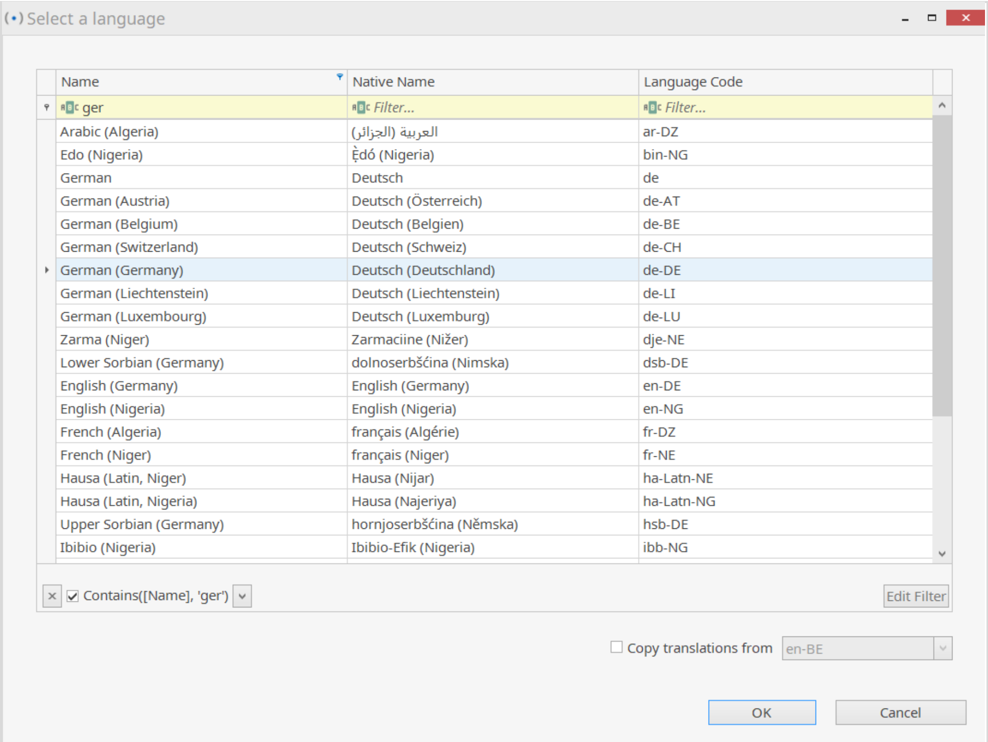


2. Select **Add Language**.



The **Select Language** window appears.

3. Select the language you want to add and click **OK**.



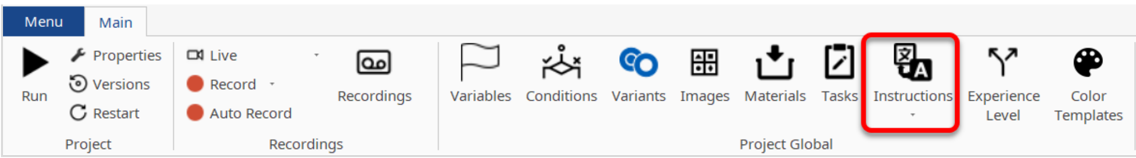
> ***Note***
>
> *If you want to prefill the text instruction with the values from another language, check the* ***Copy translations from checkbox*** *and select the language of your choice.*

The language is added. For every existing process step, an additional column is added in which you can define the translation in the new language.

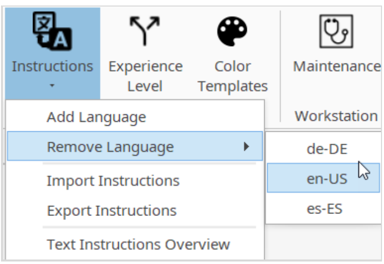
> ***Note***
>
> *For more information on adding text instructions, refer to section* ***5.5.6 Managing text and image instructions*** *on page 135.*

Removing a language

1. In the ribbon, in the **Project Global** group, click the **Instructions** button.



2. Select **Remove Language** and then the language you want to remove.

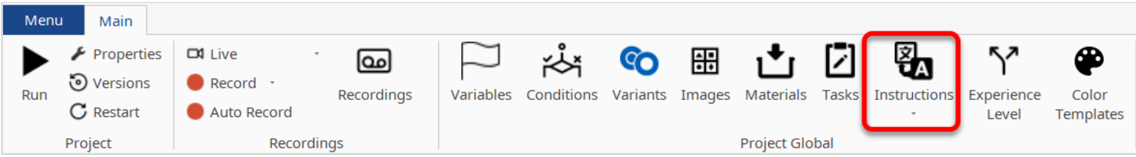


Note

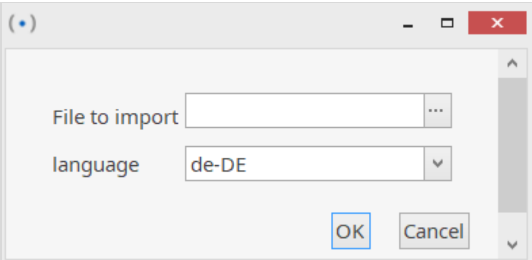
When only one language is left, you can't remove that language.

Importing text instructions

1. In the ribbon, in the **Project Global** group, click the **Instructions** button.



2. Select **Import Instructions**.
3. Select an .hdb file to import and its language and click **OK**.



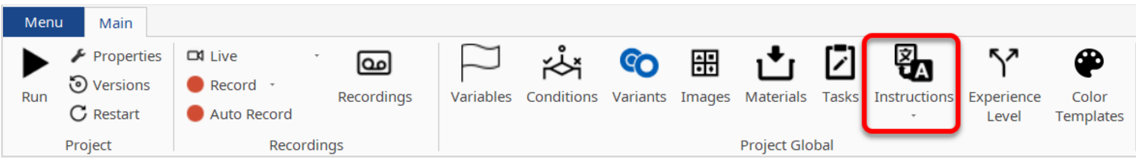
The text instructions are imported.

Note

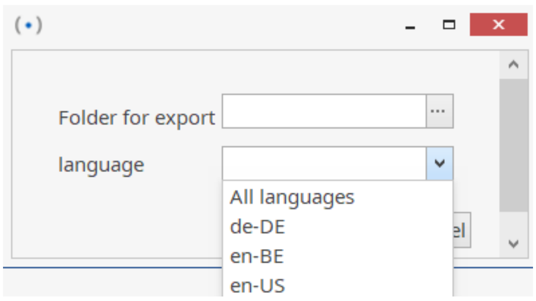
You can only import instructions for a language that is configured in your project. If the language from the .hdb file is not available yet, you first need to add it.

Exporting text instructions

1. In the ribbon, in the **Project Global** group, click the **Instructions** button.



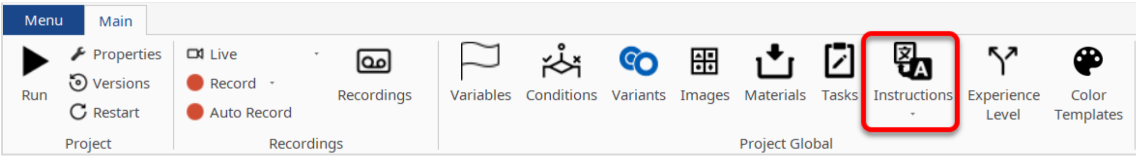
2. Select **Export Instructions**.
3. Select an export folder, the language you want to export and click **OK**.



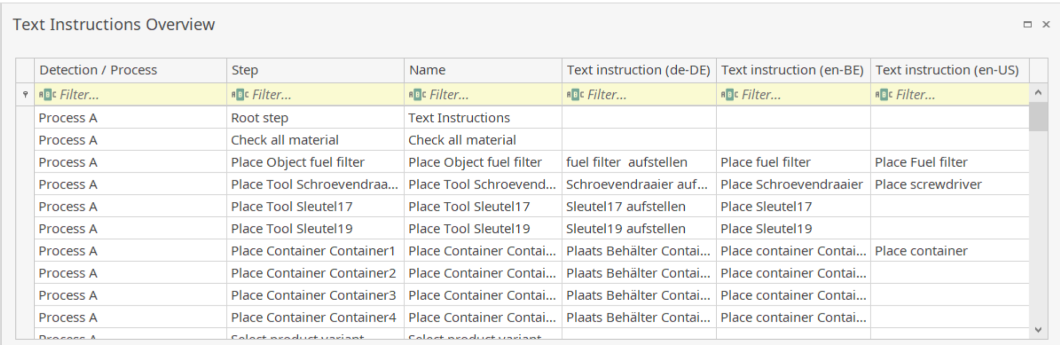
The text instructions are exported to an .hdb file.

Consulting and editing text instructions in your project

1. In the ribbon, in the **Project Global** group, click the **Instructions** button.



2. Select **Text Instructions Overview**.
   An overview of all process steps and their instructions in all configured languages appears. You can directly edit them in this overview.



Text Instructions Overview

Detection / Process	Step	Name	Text instruction (de-DE)	Text instruction (en-BE)	Text instruction (en-US)
Filter...	Filter...	Filter...	Filter...	Filter...	Filter...
Process A	Root step	Text Instructions			
Process A	Check all material	Check all material			
Process A	Place Object fuel filter	Place Object fuel filter	fuel filter aufstellen	Place fuel filter	Place Fuel filter
Process A	Place Tool Schroevendraa...	Place Tool Schroevend...	Schroevendraaier auf...	Place Schroevendraaier	Place screwdriver
Process A	Place Tool Sleutel17	Place Tool Sleutel17	Sleutel17 aufstellen	Place Sleutel17	
Process A	Place Tool Sleutel19	Place Tool Sleutel19	Sleutel19 aufstellen	Place Sleutel19	
Process A	Place Container Container1	Place Container Contai...	Plaats Behälter Contai...	Place container Contai...	Place container
Process A	Place Container Container2	Place Container Contai...	Plaats Behälter Contai...	Place container Contai...	
Process A	Place Container Container3	Place Container Contai...	Plaats Behälter Contai...	Place container Contai...	
Process A	Place Container Container4	Place Container Contai...	Plaats Behälter Contai...	Place container Contai...	

16. User management

16.1 About user management

Access to the Arkite software is granted through a username and password. That can either be a local user created in the Arkite software, or an Active Directory user if Active Directory authentication has been configured.

The actions that can be performed in the Arkite software depend on the role that is assigned to the user or the Active Directory User Group. A role is a set of permissions.

> ***Important***
>
> *Users and roles are managed on server level. This means that a user that has been configured can log in to the Arkite software on each workstation that is connected to the server.*

16.2 Managing Authentication Settings

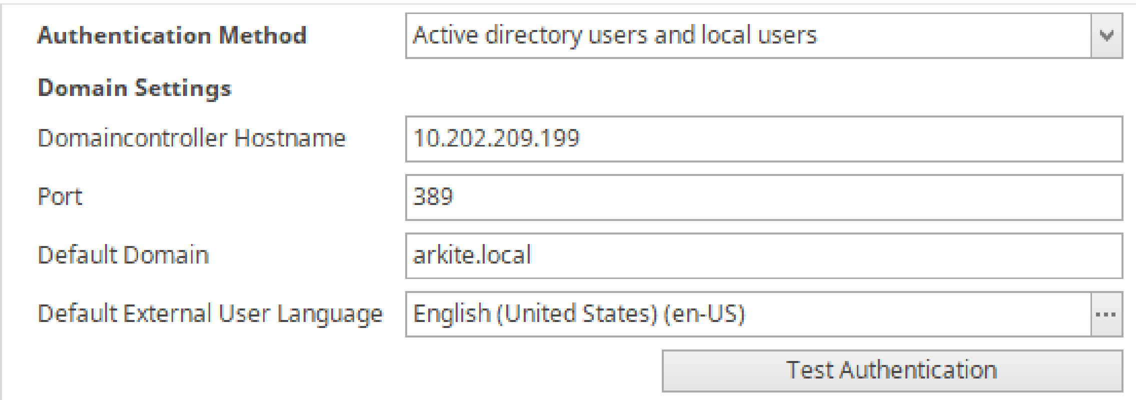
What?

By default, users are created and managed locally in the Arkite software. However, you can also allow authentication with an Active Directory user.

Configuring Active Directory Authentication

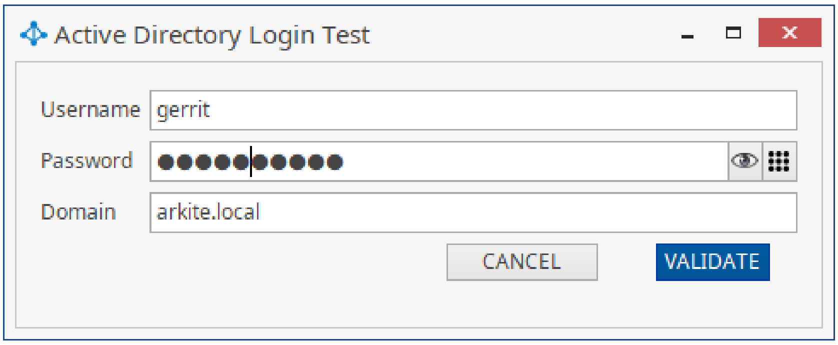
To configure Active Directory, execute the steps below:

1. Click the **Menu** tab.
2. Click **Users** and then under **Authentication** click **Settings**.
3. From the **Authentication Method** dropdown list, select **Active directory users and local users**.
4. Specify the domaincontroller hostname, port, default domain and default external user language for your Active Directory setup.

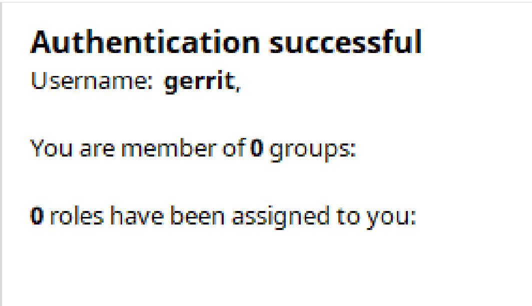


> ***Note***
>
> *If a language is assigned to the user in Active Directory, this language will also be used in the Arkite software. If no language is assigned in Active Directory, the language specified in the **Default External User Language** field will be used.*

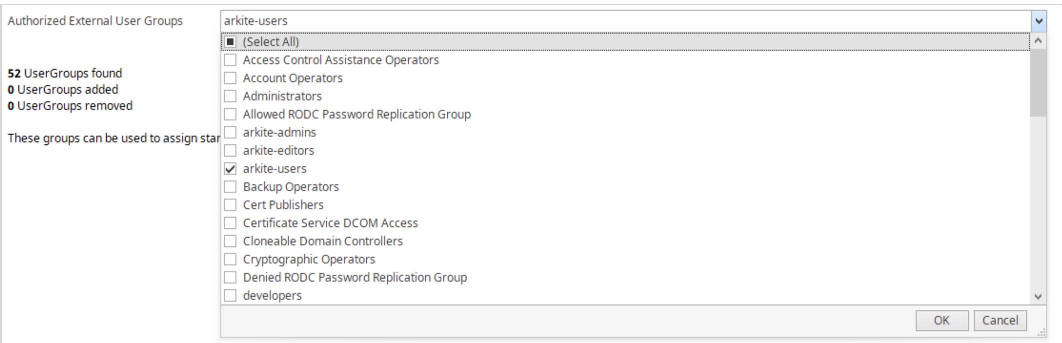
5. To test the connection, click the **Test Authentication** button.
6. Provide an Active Directory user name and password and click **Validate**.



If the connection was successful, an authentication successful message will be shown.



7. To fetch the user groups from your Active Directory, click **Update External User Groups**.
8. Provide an Active Directory user name and password and click **Validate**.
   A message is shown with the number of found user groups.
9. In the **Authorized External User Groups** list, indicate the user groups you want to grant access to the Arkite software and click **OK**.



Result

All Active Directory users that are part of an authorized user group will be able to login to the Arkite software using their Active Directory login and password.

By default, an authorized Active Directory user will have read-only access to projects in status **Production**, similar to the Standard User (Operator)**.** To grant more permissions to the Active Directory user, assign the external user group to a role.

16.3 Managing roles

Default roles

By default, four roles are configured in the Arkite software:

- **System Admin:**
  users with this role can edit settings and manage projects and users.
- **User Manager:**
  users with this role can configure users and roles.
- **Editor**:
  users with this role can edit projects and move them from **Draft** to **Review** and from **Review** to **Production** (if no approval is required).
- **Approver**:
  users with this role can do the same as editors, but can also approve or reject a project.

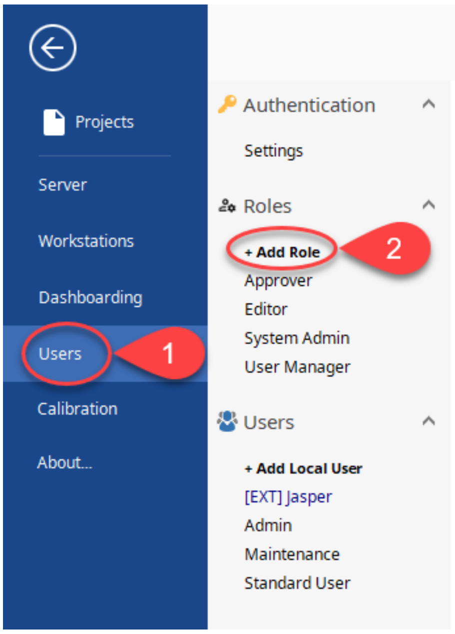
The table below gives an overview of the permissions of these default roles:

Permission	Description	System Admin	User Manager	Editor	Approver
Manage Users	Add, modify and delete users and roles. Assign roles to users and permissions to roles.	✓	✓	-	-
Edit Settings	Edit settings of the server.	✓	✓	-	-
Manage Projects	Add, modify and delete projects	✓	-	✓	✓
Approve Projects	Approve or reject a project	-	-	-	✓

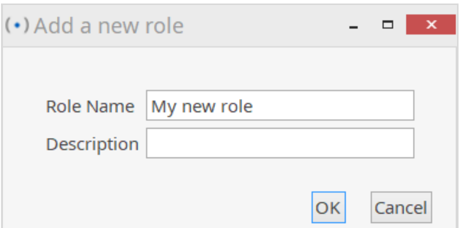
Adding roles

To add a role, execute the steps below:

1. Click the **Menu** tab.
2. Click **Users** and then click **Add Role**.



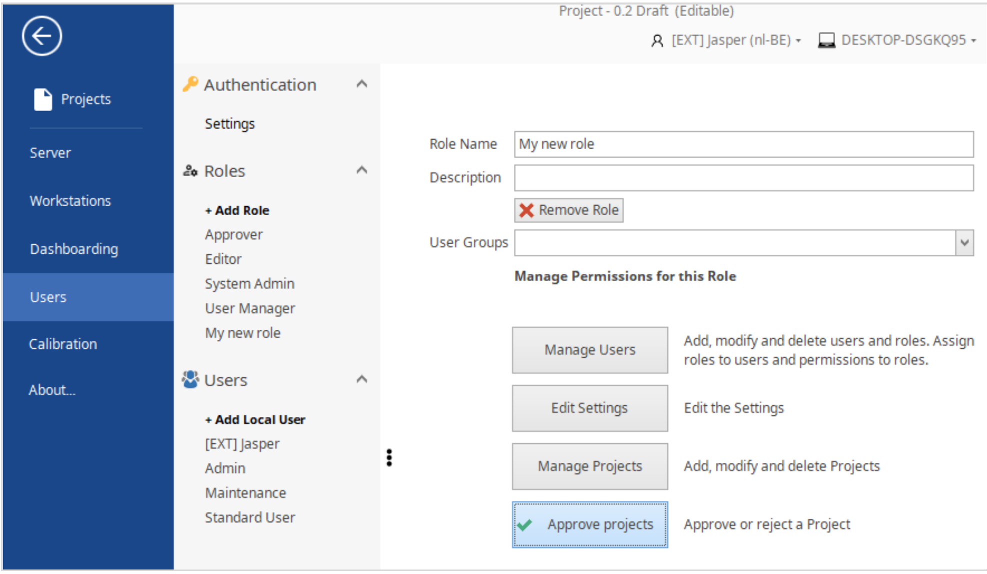
3. Provide a name and (optionally) a description for the new role and click **OK**.



4. (optional) If Active Directory configuration has been configured, you can assign the role to one or more Active Directory user groups in the **User Groups** dropdown list.

Note
*An Active Directory user will only have access to the software if the selected user group or another user group he is part of is marked as an **Authorized External User Group** in the authentication **Settings**.*

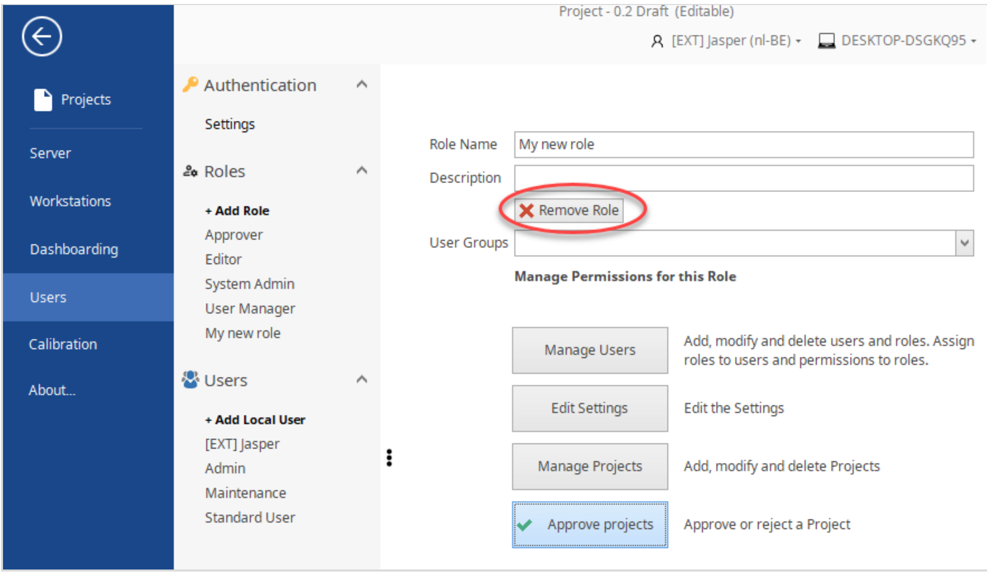
5. Click the permissions you want to assign to the role.
   In the example below, only the permission **Approve Projects** has been assigned to the role.



Editing roles

To edit an existing role, execute the steps below:

1. Click the **Menu** tab.
2. Click **Users.**
3. Click the role you want to edit.
4. Update the name, description and permissions of the role as required.
   If you want to delete the role, click the **Remove Role** button.



Note

The default roles cannot be removed nor edited. This way, you cannot accidentally lock yourself out of the application.

16.4 Managing local users

Default local users

By default, three local users are configured in the Arkite software:

- **Admin:**
  Can do everything, except approving or rejecting projects.
- **Maintenance:**
  Can do everything, except approving or rejecting projects. This user is only to be used by Arkite staff.
- **Standard User:**
  Has read-only access to projects in status **Production**. No password nor roles have been assigned to this user.

  Note

  If you have configured Active Directory authorization, be aware that by default it is still possible to log in with this user and get read-only access to the software. If you do not want this, you should set a password for the Standard User.

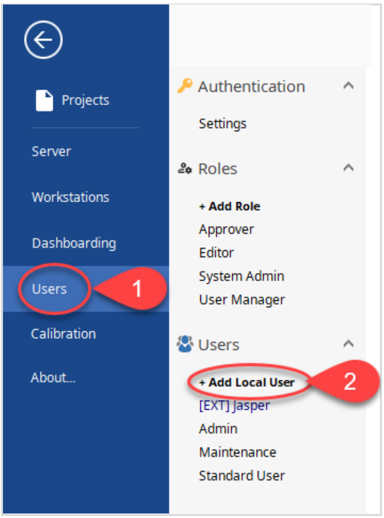
The table below gives an overview of the roles of these default local users:

Role	Admin	Maintenance	Standard User
System Admin	✓	✓	-
User Manager	✓	✓	-
Editor	-	-	-
Approver	-	-	-

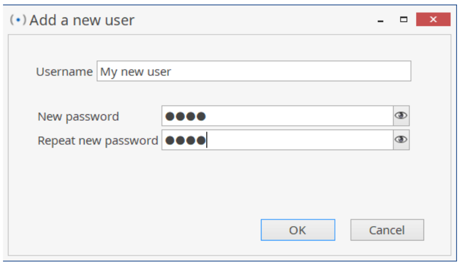
Adding local users

To add a user, execute the steps below:

1. Click the **Menu** tab.
2. Click **Users** and then click **Add Local User**.



3. Provide a username and (optionally) a password for the new user and click **OK**.



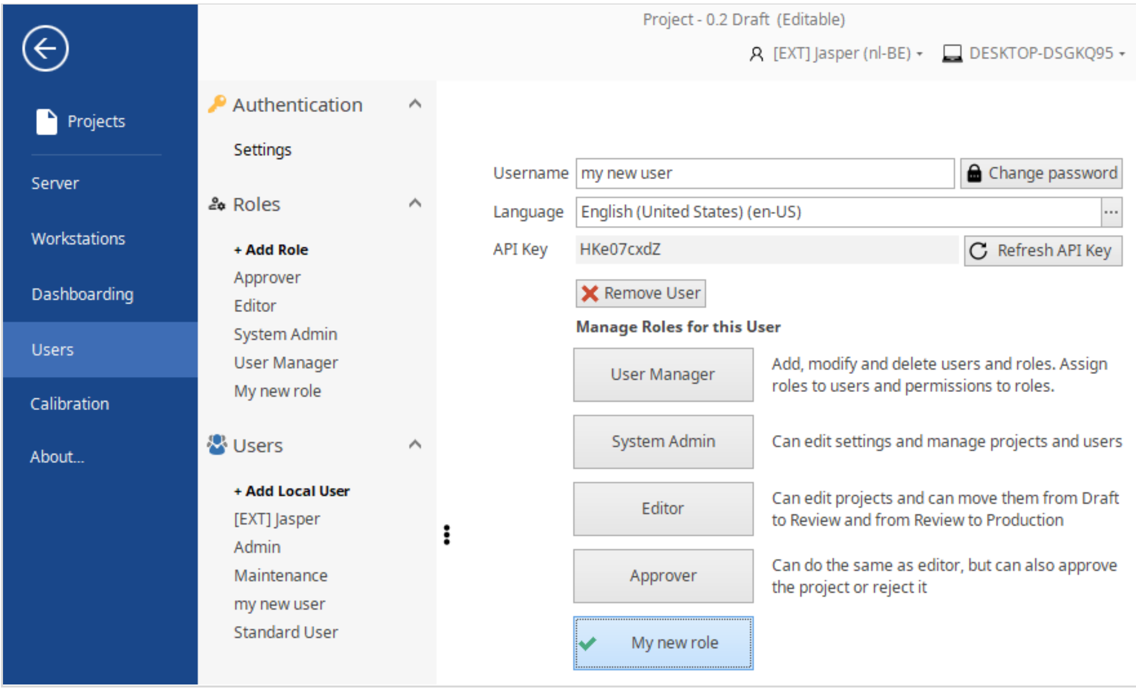
4. Select a language for the user.
   Text instructions from all projects will be shown in this language when this user is logged in.

   Note

   *If no instructions are available in the user's language, the system will first check if there are instructions available in a language with the **same primary language subtag**. If this is not the case, the system will fall back to **en-US** as a default language.*

   *For example, if user's language is **fr-BE** and no instructions are available in that language, instructions will be shown in **fr-FR** if available. Otherwise, **en-US** will be used.*

5. Click the roles you want to assign to the user.
   In the example below, only the new role we created in the section above has been assigned to the user.



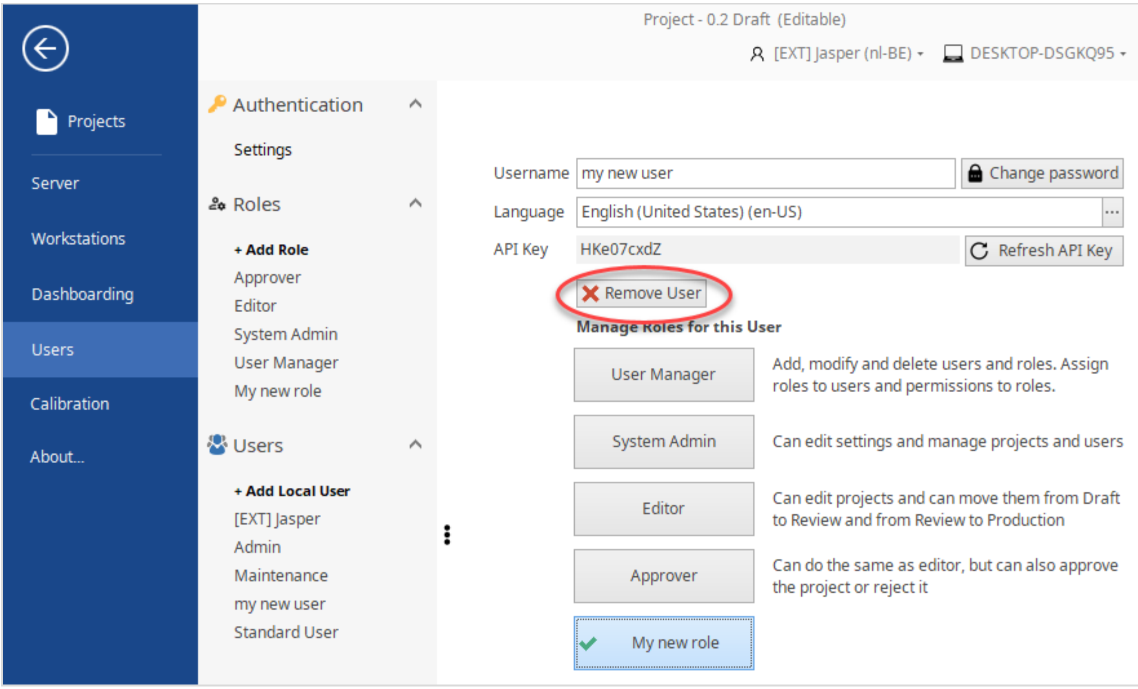
Note

*For every user that is added, the system automatically generates an API key. This key is used to authenticate when accessing the Arkite API. You can let the system generate a new key by clicking the **Refresh API Key** button. When you do so, existing functionality will not work anymore though.*

Editing users

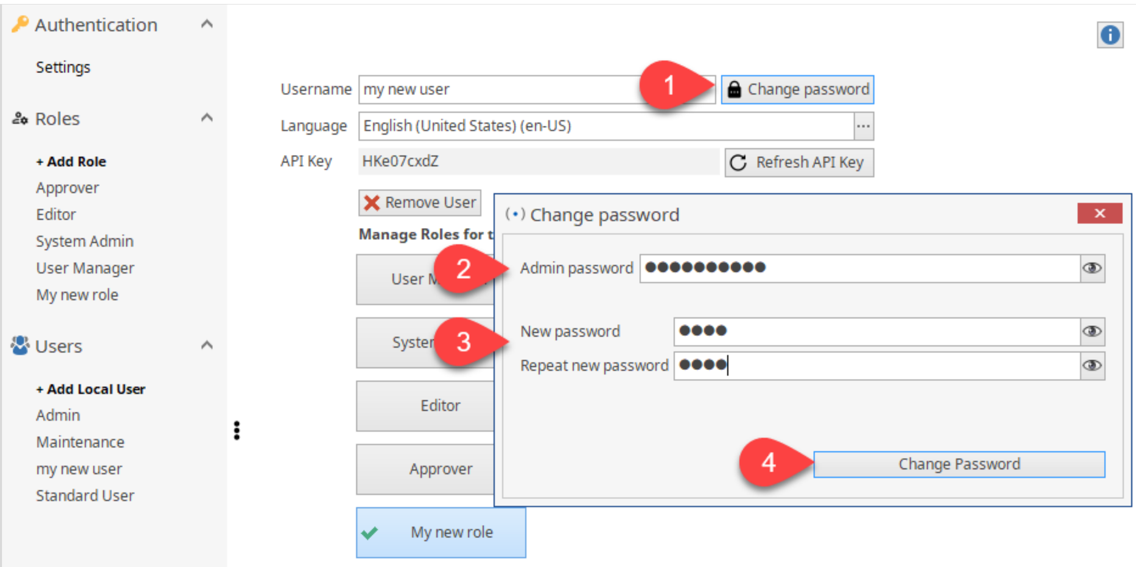
To edit an existing user, execute the steps below:

1. Click the **Menu** tab.
2. Click **Users**.
3. Click the user you want to edit.
4. Update the name, language and roles of the user as required.
   If you want to delete the user, click the **Remove User** button.



To change a user's password:

- click the **Change Password** button.
- Provide your own password, the new password for the user, and click **Change Password**.



Note

- *You can click the 👁 button to make the password you've entered visible and verify if you haven't made any typing mistakes.*
- *You can't remove any of the default users.*
- *It is also not possible to edit the roles of the **Admin** user. This way, you cannot accidentally lock yourself out of the application.*

17. The Arkite API

17.1 About the Arkite API

The Arkite software has a flexible Application Programming Interface (API) that allows swift communication with other software applications in the software ecosystem of the manufacturer like MES and ERP systems.

This API can be used to push data from MES into the Arkite software in order to enable the on-demand flexible production. This integration will avoid stock pile-up and can benefit from the fact that the Arkite software is flexible to be programmed to deal with product variants on the fly.

It contains rest calls for retrieving and modifying project data, as well as a number of calls to run operations on a currently running workstation.

17.2 API examples

The Arkite API can be used for example to:

- Schedule tasks on a workstation when an order is received by your ERP system
- Load a project on a specific workstation when a new workday starts
- Update the container content from your WMS system
- Get the status of a detection
- Add a variant
- ...

Note
*For most entities in the Arkite software, an object ID is displayed next to the label of the **Name** property on the **Properties** pane. In the **Workstations** overview, the workstation's object ID is shown in the **Workstation ID** column. You can copy it to use it when working with the Arkite API.*

17.3 More information

Extended documentation of the Arkite API can be found on SwaggerHub:

https://app.swaggerhub.com/apis-docs/Arkite/ArkiteAPI/2023.3

For testing, you can use the Open API Tester. To access this page, click the ? **Help** button in the ribbon of the software and select **Open API Tester**.