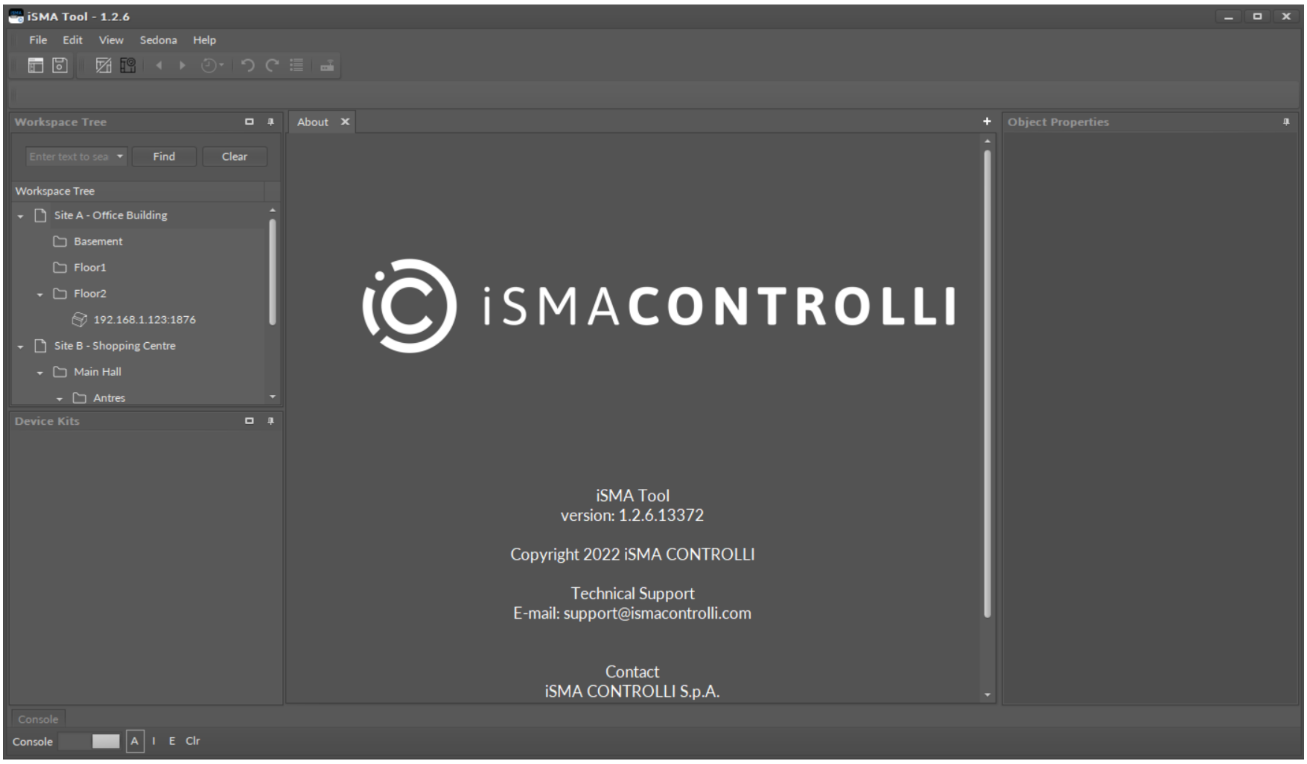
iSMA Tool

User Manual



iSMA Tool
version: 1.2.6.13372

Copyright 2022 iSMA CONTROLLI

Technical Support
E-mail: support@ismacontrolli.com

Contact
iSMA CONTROLLI S.p.A.

Table of Contents

1 Introduction

As a significant part of an end-to-end iSMA solution, the iSMA Tool gives customer a convenient way to create and manage custom applications for a Sedona-based iSMA controller.

The iSMA Tool now covers all requirements to create a perfect application: a wire sheet for convenient visual programming, property sheets for details, kit management, real-time monitoring of system states and slot values, logs and historical data, and deployment and backup.

1.1 Revision History

Rev.	Date	Description
1.1	1 May 2018	First edition
1.2	1 Oct 2018	Updated to iSMA Tool ver. 1.1.1
1.3	6 Dec 2018	Updated to iSMA Tool ver. 1.2.0
1.4	21 Jan 2019	Updated to iSMA Tool ver. 1.2.1
1.5	24 Jul 2019	Updated to iSMA Tool ver. 1.2.2
1.6	21 Apr 2022	Updated to iSMA Tool ver. 1.2.6 Rebranded

Table 1. Revision history

2 About

The iSMA Tool allows for programming, parameter setting, and building of applications in the iSMA-B-AAC20 controller. Using proper functions implemented in the iSMA Tool, the user can build his own application from scratch or edit an existing application working directly on a connected device. The iSMA Tool also allows for working with iSMA-B-FCU controllers.

The iSMA Tool is addressed to every user working with the Sedona controllers on the daily basis, i.e., distributors, integrators, engineers, and maintenance technicians.

The iSMA Tool is compatible with devices manufactured by iSMA CONTROLLI. Working with devices manufactured by third parties, and programming them, is not possible. There is a possibility of unlocking the iSMA Tool for an individual customer, including handling of additional controllers. For this purpose contact the iSMA CONTROLLI Support Team at ismacontrolli.com.

2.1 Naming and Definitions

A proper understanding of the terms used hereafter will allow the iSMA Tool user to adopt the described issues in a more proficient way.

2.1.1 Basic Definitions

- iSMA Tool: the name of a computer program this document applies to;
- component: a single element of an application working in the SVM (Sedona Virtual Machine) environment. The component is either a function block with a pre-defined task (algorithm), which requires specific parameters and logical connections with other components to work and exchange data properly, or a container (folder to segregate other components);
- kit: a set of components with similar functionalities or working in the same area, e.g., components handling controller's inputs and outputs. In the iSMA Tool, the Kits and their components are shown in a Device Kits window;
- link: a logical connection between components: such connection allows two components to send data between each other. The graphical representation of a link in the Wire Sheet view is a line connecting components;
- slot: an element of the component which defines the parameter along with its name, value, unit, and range. Each component has a slot named Meta and a number of other editable or read-only slots available to the user. Slots are not visible in all views in order to avoid overwhelming the user with too much information. The views dedicated to work with the slots are: the Property Sheet, Slot Sheet, and Wire Sheet;
- selection: a user's action of choosing any component or link by clicking it. The chosen object becomes highlighted graphically. In the Wire Sheet view the component's selection is illustrated by surrounding a component with an orange frame, and a link's selection is illustrated by changing its color to orange. The selected element is an object to which the commands such as copying, deleting, or moving will relate;
- marking: a state relating to the component or link connected with the selected object. Marking means that a specific component or link is in correlation with a selected component/link. In the Wire Sheet view a green color is used to mark a component or link, which allows the user to easily notice which components/links will be affected by

the action taken on the selected object, e.g., after deleting a selected component/link the marked object will lose its correlation;
- Workspace Tree: one of the basic windows provided by the iSMA Tool. The window shows the connected devices along with their components. Additionally, the devices may be assigned hierarchically to folders, and folders to projects;
- Device Kits: one of the basic windows provided by the iSMA Tool. It is a kit library along with components assigned to them. The content of the Device Kits is dynamically adjusted to the type of connected device;
- Main screen: a main workspace of the iSMA Tool, where the views of editable components are displayed on tabs;
- Object Properties window: a window showing information regarding a selected element allowing to preview and change some of the selected element's parameters without closing the current view;
- Console: a window where the messages regarding the operation of the iSMA Tool program are displayed.

2.1.2 Component Views

A component shows its data on views assigned to it. A list below shows four basic views, which present data about the same component in different ways, making the work with components easier. Some components, such as a component showing historical data on graphs or a component allowing the creation of complex work schedules, have also individual views (apart from the standard ones), which are not included in the list below.

- Wire Sheet: a view showing programming relations between the components in a graphical way. The view shows logical relations between the linked components. This view makes building an application logic easier, because the connections between the components are easy to enter and analyze. It is the view most commonly used on the application programming stage;
- Property Sheet: a tabular form of showing parameters: component slots. The user can see not only the slot value, but also its units and range. The view allows to change the slot value if it has the change value rights. The Property Sheet is often used at the start-up stage, when the settings of many parameters (slots) in components need to be appointed;
- Slot Sheet: a component view showing all slots, including the hidden ones. In a tabular form it allows to read the name, type, and parameters (Facets) of the slot.

3 Installation and Updates

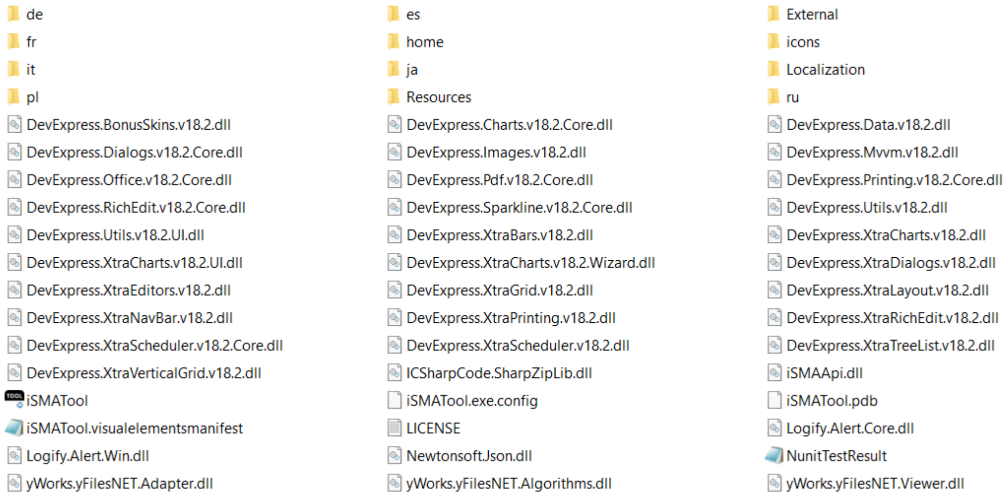
3.1 Installation

The iSMA Tool is a software created for modern Microsoft Windows system, such as Windows 10. The oldest supported version of the operating system is Windows 7. The iSMA Tool is delivered as a compressed folder, which needs to be extracted in a chosen location on a hard drive, unless the access to the extracted folder is restricted by the system (e.g., Program Files is not a recommended location).

In order to download the iSMA Tool Software Bundle, which includes all files necessary to run the program efficiently (a zipped file iSMATool_Vx.x.x.zip), go to the iSMA CONTROLLI web page ismacontrolli.com and to the iSMA Tool/Software Bundle folder.

Extracting the zipped package reveals the folders and additional files described below. In order to run the iSMA Tool, open the iSMATool.exe file.

In order to download the iSMA Tool Software Bundle, which includes all files necessary to run the program efficiently (a zipped file iSMATool_Vx.x.x.zip), go to the iSMA CONTROLLI web page ismacontrolli.com and to the iSMA Tool/Software Bundle folder.



de
es
External
fr
home
icons
it
ja
Localization
pl
Resources
ru
DevExpress.BonusSkins.v18.2.dll
DevExpress.Charts.v18.2.Core.dll
DevExpress.Data.v18.2.dll
DevExpress.Dialogs.v18.2.Core.dll
DevExpress.Images.v18.2.dll
DevExpress.Mvvm.v18.2.dll
DevExpress.Office.v18.2.Core.dll
DevExpress.Pdf.v18.2.Core.dll
DevExpress.Printing.v18.2.Core.dll
DevExpress.RichEdit.v18.2.Core.dll
DevExpress.Sparkline.v18.2.Core.dll
DevExpress.Utils.v18.2.dll
DevExpress.Utils.v18.2.UI.dll
DevExpress.XtraBars.v18.2.dll
DevExpress.XtraCharts.v18.2.dll
DevExpress.XtraCharts.v18.2.UI.dll
DevExpress.XtraCharts.v18.2.Wizard.dll
DevExpress.XtraDialogs.v18.2.dll
DevExpress.XtraEditors.v18.2.dll
DevExpress.XtraGrid.v18.2.dll
DevExpress.XtraLayout.v18.2.dll
DevExpress.XtraNavBar.v18.2.dll
DevExpress.XtraPrinting.v18.2.dll
DevExpress.XtraRichEdit.v18.2.dll
DevExpress.XtraScheduler.v18.2.Core.dll
DevExpress.XtraScheduler.v18.2.dll
DevExpress.XtraTreeList.v18.2.dll
DevExpress.XtraVerticalGrid.v18.2.dll
ICSharpCode.SharpZipLib.dll
iSMAApi.dll
iSMATool
iSMATool.exe.config
iSMATool.pdb
iSMATool.visualelementsmanifest
LICENSE
Logify.Alert.Core.dll
Logify.Alert.Win.dll
Newtonsoft.Json.dll
NunitTestResult
yWorks.yFilesNET.Adapter.dll
yWorks.yFilesNET.Algorithms.dll
yWorks.yFilesNET.Viewer.dll

Figure 1. Folders and files after extracting

The extracted folders have the following functions:

- Config: a folder containing a record of user's individual settings regarding windows location and other iSMA Tool work settings, such as a language chosen for the iSMA Tool interface.
- External: a folder containing an API .dll file.
- home: a folder where all the data created by user are saved, i.e., device backups, applications, etc. It is also a folder where the kits library, available the in iSMA Tool, is located.
- icons: a folder with graphical files such as the iSMA Tool interface icons.
- Localization: a folder with the text files providing the iSMA Tool language sources.
- log: a folder, where the logs of the iSMA Tool, which also appeared in Console window, are saved. When contacting iSMA CONTROLLI technical support, it is advised to copy the last file with logs form that folder
- de, es, fr, it, ja, pl, Resources, ru: folders with system libraries.

To properly install and work with the iSMA Tool, the computer must meet the following minimal requirements:

- Processor (CPU): Intel Core i3-3xxx or equivalent;
- Memory: 4GB RAM;
- Storage: 50 GB internal hard driver;
- Ethernet 100 Mbit or 1Gbit NIC;
- MS Windows 7 (recommended MS Windows 10);
- .NET Framework 4.6.2 or higher.

Warning!

When the iSMA Tool is being run for the first time, it asks to accept the EULA license. The license must be accepted to run the program. Failure to do so closes the iSMA Tool.

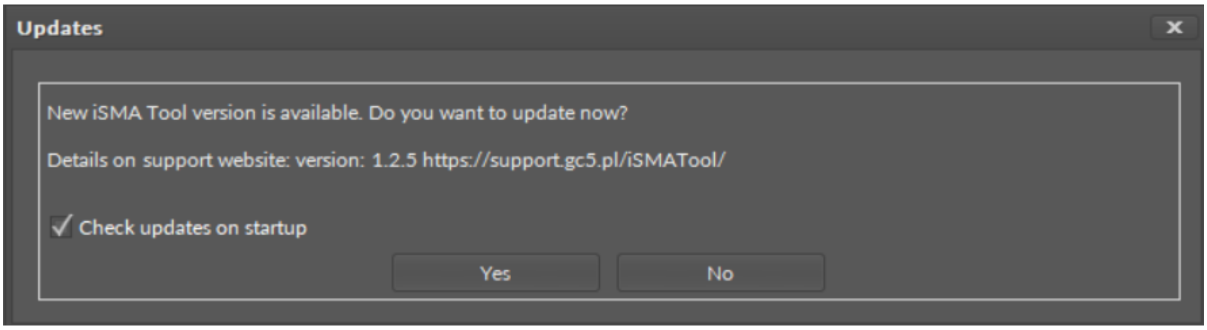
Worth to Notice

In order to make sure that the iSMA Tool is always up to date, it needs to be run periodically at least once a month, on a computer connected to the Internet for about an hour, depending on the data transfer rate. It enables the iSMA Tool to automatically download the latest data, such as kits and updates.

The iSMA Tool is a portable software. It is transferable and it can be installed on a portable data storage device, such as a USB memory stick. It allows the iSMA Tool to be run directly from a portable data storage device on any PC, including offline ones.

3.2 Updating

The iSMA Tool has a built-in mechanism notifying the user when a new version of software is available on the iSMA CONTROLLI server. Every time the iSMA Tool starts up and it is connected to the Internet, the software checks the availability of a new version. If the new version is available, a pop-up widow, similar to the one shown below, will appear.



Updates

New iSMA Tool version is available. Do you want to update now?

Details on support website: version: 1.2.5 https://support.gc5.pl/iSMATool/

☑ Check updates on startup

Figure 2. Updates window

The options in the Updates window are the following:

- Yes: Allows downloading a new version of the iSMA Tool by opening an appropriate location in the user's default web browser. The new version of the iSMA Tool is downloaded as a zipped file saved locally on the computer’s hard drive and then extracted, overwriting the current iSMA Tool version.

- No: Closes the iSMA Tool updates notification window. The window will appear again after restarting the iSMA Tool.

Warning!

Before overwriting the current version of the iSMA Tool, the software needs to be closed and the user has to make sure that all important data (kits, applications, settings) are saved in dedicated folders, home and Config. These folders will not be overwritten or deleted while extracting/overwriting the current iSMA Tool version with an updated one, preserving everything the user has created.

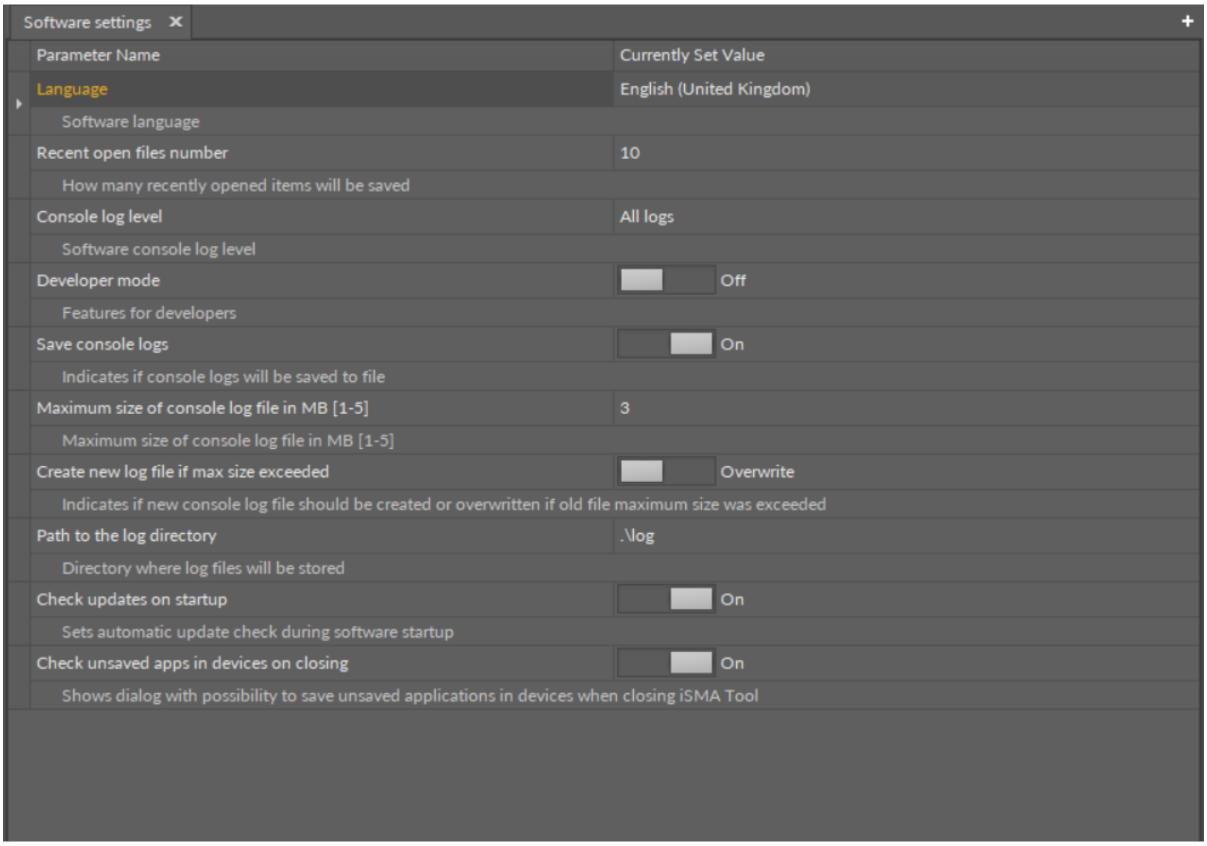
The automatic checking mechanism, described above, may be turned on or off in the iSMA Tool settings described in Chapter 10 iSMA Tool Settings. When the checking mechanism is turned on and there are no new versions of the iSMA Tool, the system will display an appropriate notification: No updates for iSMA Tool on the console.

There is also a manual way of checking software updates availability. In order to do so, initiate the menu Check updates, available in upper Help menu. After initiating this option, the iSMA Tool will check the availability of new versions and display an appropriate notification. If a new update is available, a pop-up window, described above, will appear.

4 Settings

4.1 Software Settings

The iSMA Tool allows to adjust individual settings for a better fit of the working environment to the user's needs. Such settings include, among others, the language interface, number of remembered or recently opened files of the Workspace Tree, and Console settings. The settings are available in the main menu File-> Settings.



Parameter Name	Currently Set Value
Language	English (United Kingdom)
Software language	
Recent open files number	10
How many recently opened items will be saved	
Console log level	All logs
Software console log level	
Developer mode	Off
Features for developers	
Save console logs	On
Indicates if console logs will be saved to file	
Maximum size of console log file in MB [1-5]	3
Maximum size of console log file in MB [1-5]	
Create new log file if max size exceeded	Overwrite
Indicates if new console log file should be created or overwritten if old file maximum size was exceeded	
Path to the log directory	.\log
Directory where log files will be stored	
Check updates on startup	On
Sets automatic update check during software startup	
Check unsaved apps in devices on closing	On
Shows dialog with possibility to save unsaved applications in devices when closing iSMA Tool	

Figure 3. The Software Settings

The meaning of particular settings is explained under each setting line, see the figure above. Parameters to set in the Software settings include:

- language;
- number of recently opened files;
- Console log level;
- turning on/off developer mode;
- saving logs to a file;
- size of Console log file;
- action upon exceeding maximum log file size;
- path to the log directory;
- checking updates on startup;
- checking unsaved apps on closing.

4.2 Language Settings

A default language of the iSMA Tool is UK English.

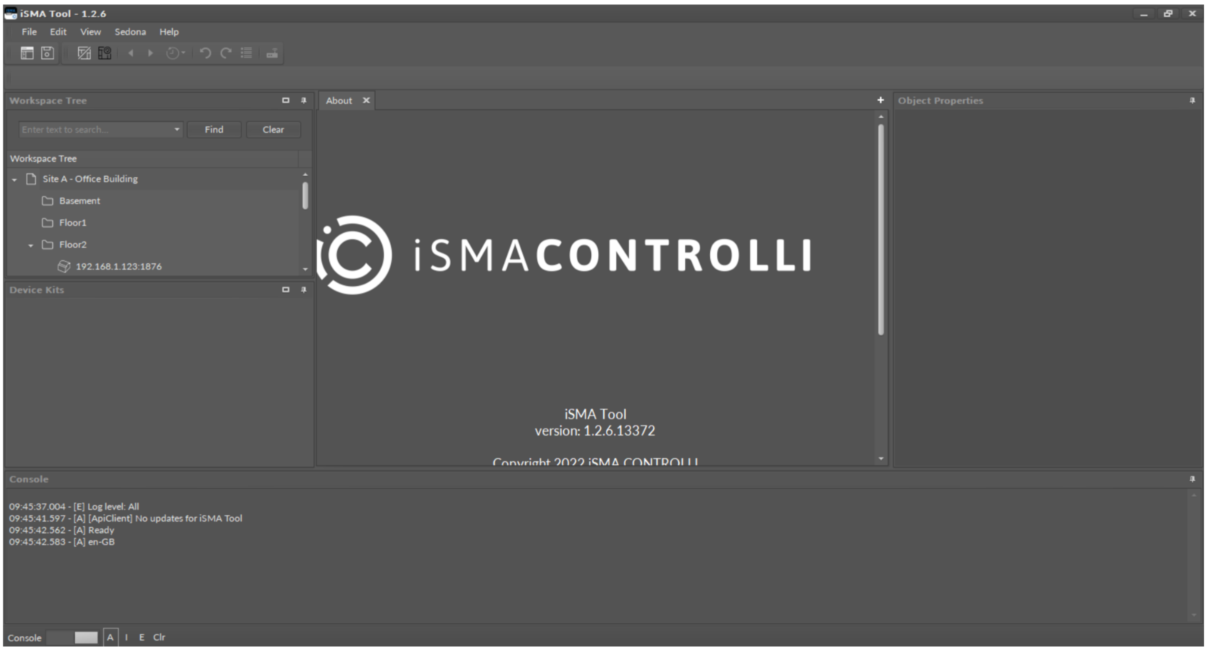
The iSMA Tool allows to change the language in the Software settings view. The language change affects a graphic interface of the iSMA Tool. After the language is changed, the program needs to be restarted.

If the language is not on the supported languages list, please contact iSMA CONTROLLI Technical Support to discuss the options (contact information is available in the About view, opened from the main menu Help -> About).

5 Navigation and Windows

5.1 iSMA Tool Windows

When the iSMA Tool is run for the first time, the iSMA Tool application is displayed. It consists of a few windows shown on the figure below.



iSMA Tool
version: 1.2.6.13372

Copyright 2022 iSMA CONTROLLI

```
09:45:37.004 - [E] Log level: All
09:45:41.597 - [A] [ApiClient] No updates for iSMA Tool
09:45:42.562 - [A] Ready
09:45:42.583 - [A] en-GB
```

Figure 4. The iSMA Tool initial window

The above look applies only during an initial activation of the iSMA Tool. Further activations and updates of the program will remember the window settings changed by the user in accordance with his preferences.

In the default settings of the iSMA Tool, the upper left side of the screen is occupied by a Workspace Tree window. Below is a Device Kits window showing current kits of the connected device. Central part of the screen is occupied by a Main screen, a space where the application's views are shown on tabs, which is discussed further. After the iSMA Tool activation the Main screen shows program information shown on the figure above. Under the Main screen there is a Console window, and on the right there is an Object Properties window.

Additionally, besides the aforementioned windows, the iSMA Tool screen shows a menu and a toolbar, a window with editable component path, and a capacity identifier.

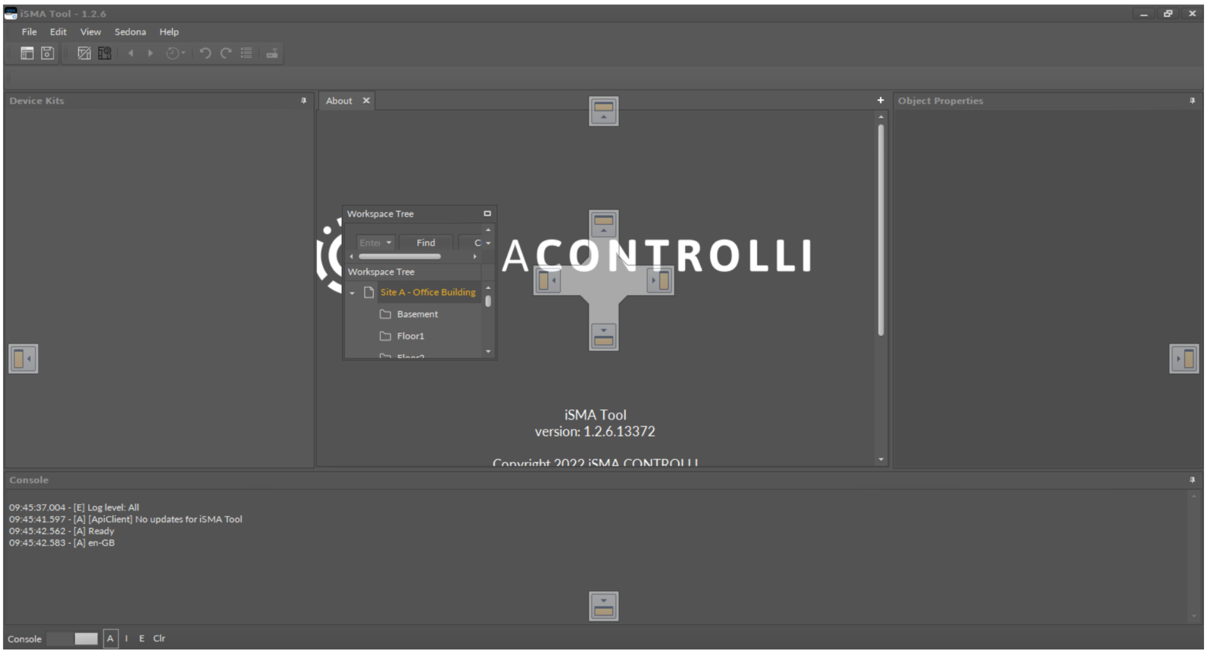
The window with an editable component path is by default set below the main menu and toolbar. It can be moved to the bottom, below the Console window, or even to the left, or right margin by changing the path text to the vertical one.

The suggested window position in the iSMA Tool can be modified freely by dragging, resizing, and fixing views relative to each other.

5.2 Windows Adjustment

The iSMA Tool allows a completely free setting of the views and windows. Free distribution allows specific windows to be visible non-stop or hide automatically. These settings apply to the windows such as the Workspace Tree, Device Kits, Object Properties, or Console windows. It is possible to move the window outside of the iSMA Tool software to other connected screen.

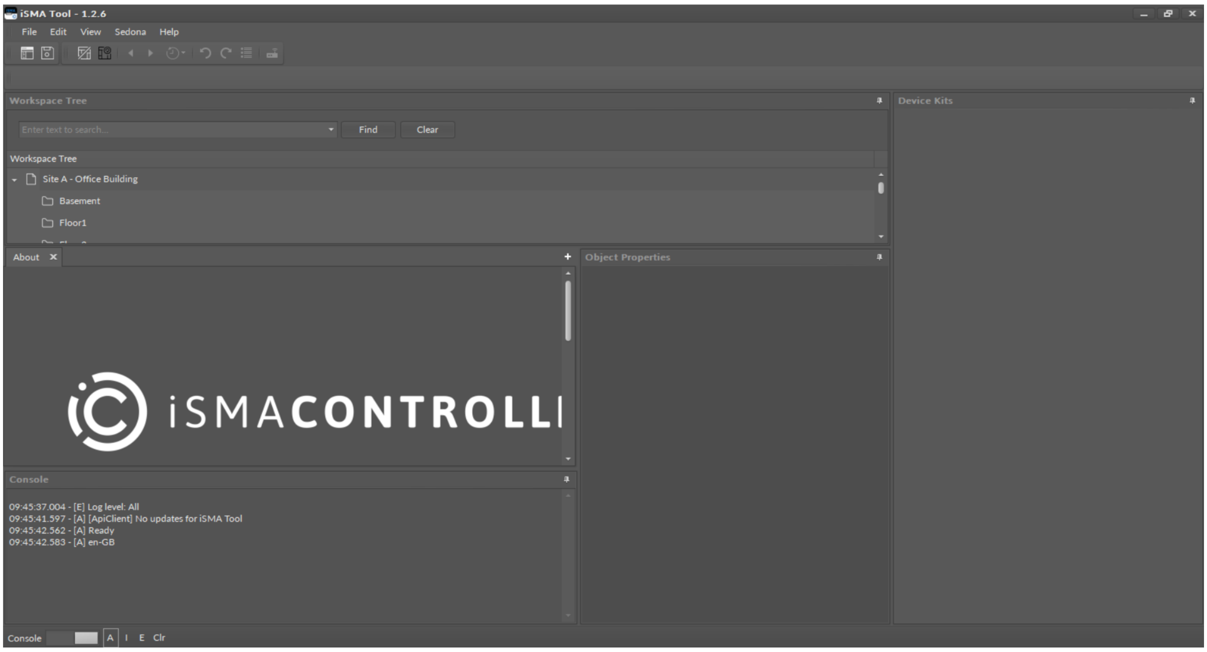
To move any window into another part of the screen it needs to be grabbed by the header and dragged into a new location. The figure below shows moving of the Object Properties window. While moving the window, special controls appear to facilitate proper placing of the window in relation to the remaining windows in the view.



```
09:45:37.004 - [E] Log level: All
09:45:41.597 - [A] [ApiClient] No updates for iSMA Tool
09:45:42.562 - [A] Ready
09:45:42.583 - [A] en-GB
```

Figure 5. Adjusting controls of the iSMA Tool windows

As a result of changing the windows position it is possible to adjust their look according to individual needs. On the figure below, the Workspace Tree window has been moved above the Main screen, and the Device Kits window has changed its position from the left side to the right.



```
09:45:37.004 - [E] Log level: All
09:45:41.597 - [A] [ApiClient] No updates for iSMA Tool
09:45:42.562 - [A] Ready
09:45:42.583 - [A] en-GB
```

Figure 6. Rearranged windows

The iSMA Tool offers a quick enlarging of the Main screen to the whole area thanks to automatic minimizing of the remaining windows. To temporarily enlarge the Main screen press the windows icon (third one from the right) on the toolbar under the main menu, as it is depicted on the figures below showing the iSMA Tool before and after the enlargement of Main screen. In order to restore the full view of all windows press again the windows icon.

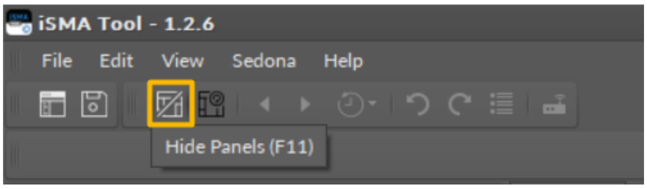
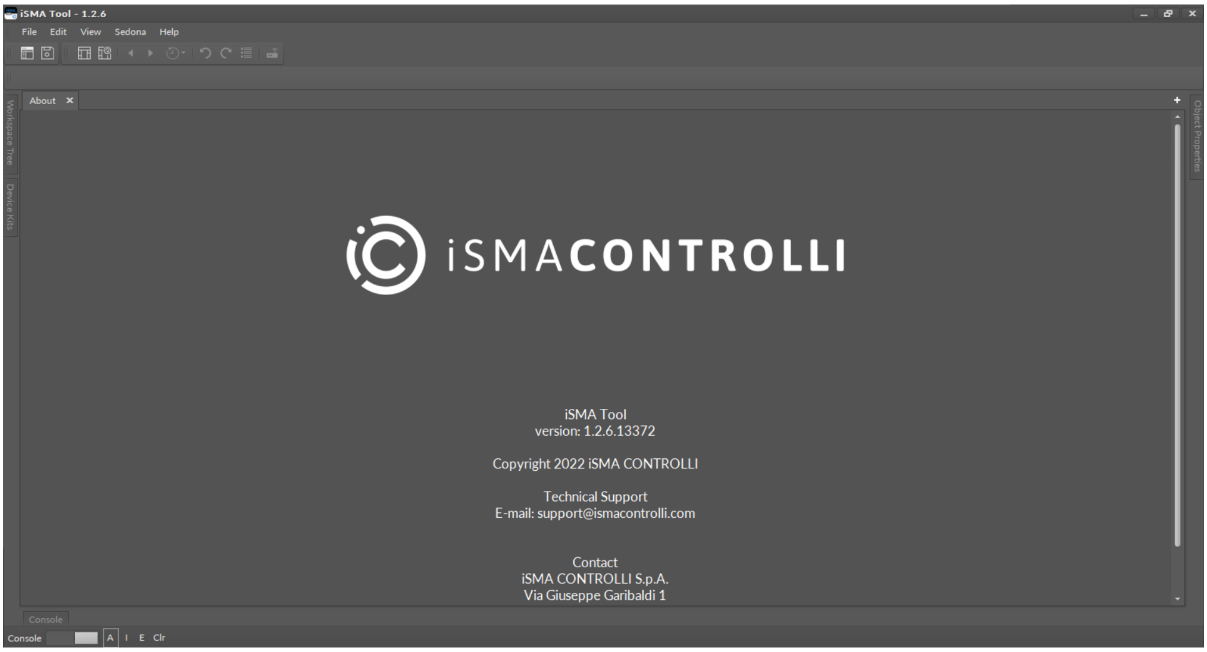


Figure 7. Hide panels button



iSMA Tool
version: 1.2.6.13372

Copyright 2022 iSMA CONTROLLI

Technical Support
E-mail: support@ismacontrolli.com

Contact
iSMA CONTROLLI S.p.A.
Via Giuseppe Garibaldi 1

Figure 8. Temporarily enlarged main screen

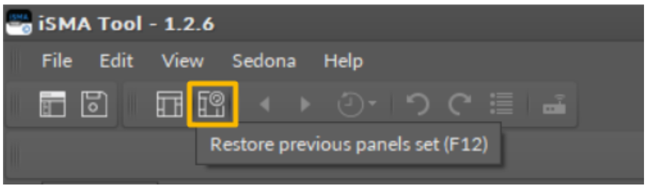
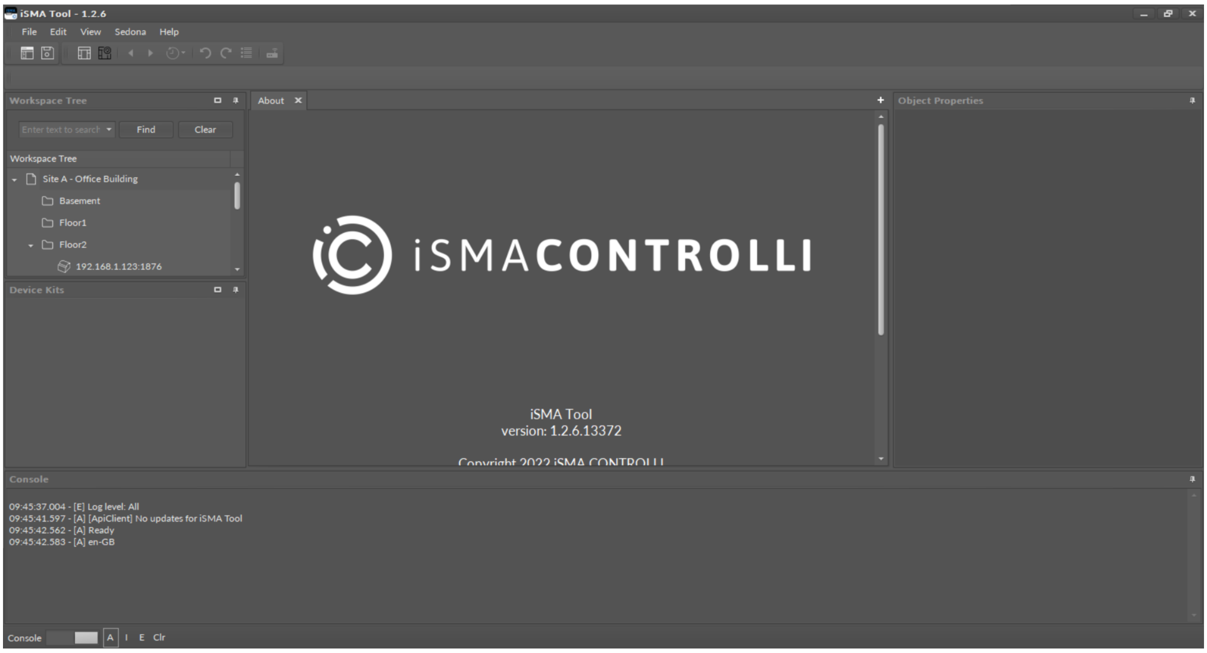


Figure 9. Restore previous panels button



iSMA Tool
version: 1.2.6.13372

Copyright 2022 iSMA CONTROLLI

```
09:45:37.004 - [E] Log level: All
09:45:41.597 - [A] [ApiClient] No updates for iSMA Tool
09:45:42.562 - [A] Ready
09:45:42.583 - [A] en-GB
```

Figure 10. Restored windows

5.3 Workspace Tree

A Workspace Tree, also known as a project tree, is a window designed to work with devices. Thanks to a hierarchical structure of the tree it is easy to switch between the devices and their components. It is also a place to add/remove the controllers and to manage components.

The iSMA Tool Workspace Tree is designed to navigate the projects and devices defined for them. For each project it is possible to define any number of folders and any number of devices inside them. It is recommended that the structure of folders, subfolders, and devices corresponds to their physical placement and/or logical subordination of controllers on an object. It makes finding a particular controller among a vast quantity of devices fast and easy. The structure of folders, subfolders, and projects mentioned above allows to properly locate a specific controller in the Workspace Tree. The hierarchical structure inside the controller is loaded directly from the controller and consists of components. There are different controller components, depending on services and functions offered by a specific device.

The Workspace Tree is saved in an .itw file in a specified location [home\Workspaces]. The user can save the Workspace Tree under a defined name in a specific location. The .itw (Workspace) file stores the whole structure of the Workspace Tree including devices names and their addresses. This allows to save many projects in one .itw file or otherwise, each individual project in a separate .itw file.

A file with the .itw extension is an .xml file and contains sensitive data, such as device IP addresses and their passwords if the user issued a command to remember them. Due to an obvious threat of exposing such data, the iSMA Tool codes sensitive data automatically, preventing their exposure to third parties.

An exemplary window of the Workspace Tree is shown on the figure below.

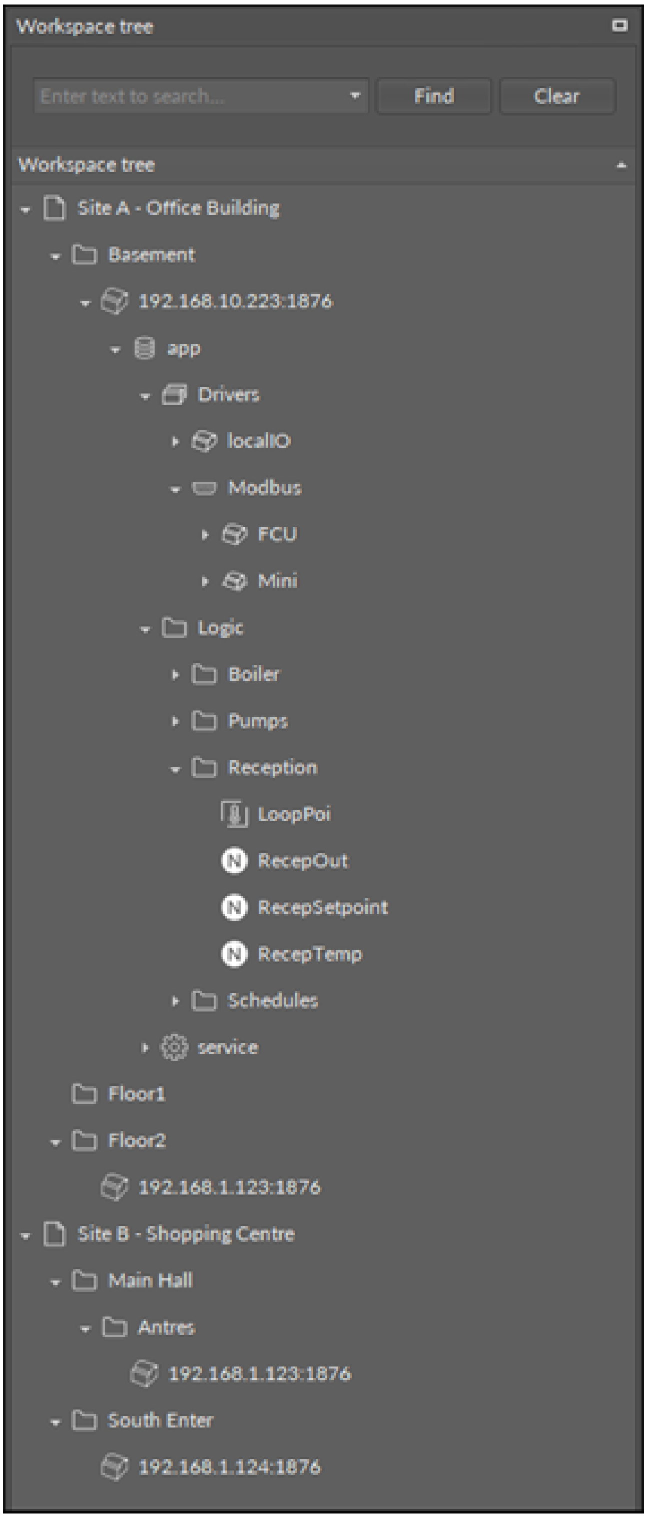


Figure 11. The Workspace Tree

5.3.1 Workspace Tree Structure

There are two independent projects on the Workspace Tree figure: Site A and Site B. Each project is a separate tree with folders and devices. Devices (objects with IP numbers) were defined in folders specifying their location such as Basement and Floor2, or Antres and South Enter. Folder Antres has been placed under the main folder–Main Hall for a better organization of a controller position.

Double-clicking on the device symbol allows to connect to the device and show its contents, which is described in details further in this instruction.

5.3.2 Editing of the Workspace Tree Elements

For folders, subfolders, and the main element–project, opening the context menu provides the following options:

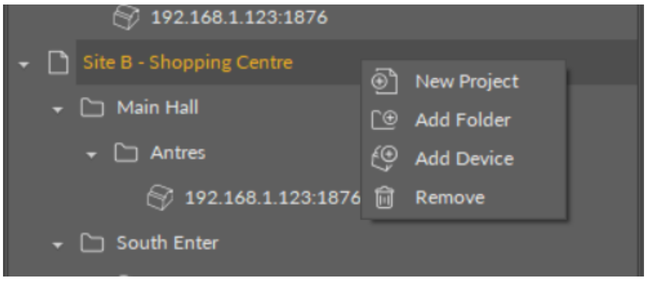


Figure 12. The Workspace Tree editing options

- New Project: creating a new project independent of the others;
- Add Folder: adding a folder under the selected folder;
- Add Device: opening a device adding window for the selected folder;
- Remove: removing the selected element: device, folder, or project from the tree.

Warning!

Removing the parent folder or project removes all subfolders and items that were structured under it, upon confirmation of operation. In the situation shown on the figure, removing the folder South Enter will result in removing the controller from the project, but the controller contents will not be deleted.

The iSMA Tool software supports UTF8 symbols in most of the places in the project. It is therefore possible to use native letters to lay out the project in a convenient way:

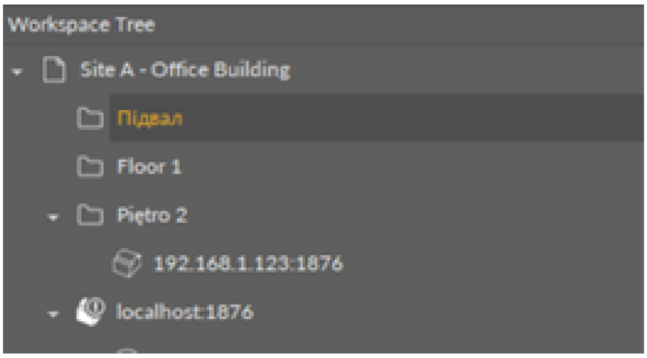
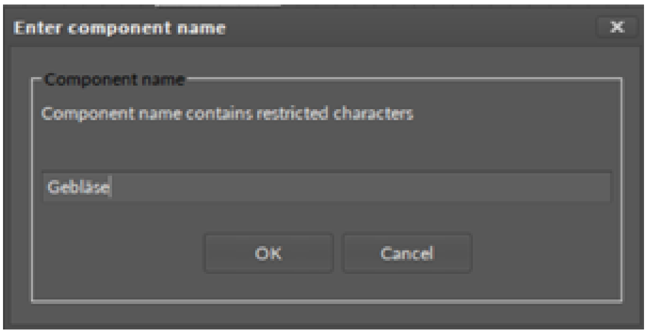


Figure 13. Example of using native letters

Note: Sedona supports a very limited set of symbols for component names, so the iSMA Tool will not allow to add one with a non-ASCII name:



Enter component name

Component name

Component name contains restricted characters

Gebläse

Figure 14. Non-ASCII characters warning

Any name can be given to a project, folder, or even device. For this purpose an object needs to be selected and clicked once more, or the function key F2 has to be used.

The device components, which can be used to build the application program, are shown on the Workspace Tree, under a controller. If a selected element of the tree has other elements under it, i.e., children elements, issuing a command to a parent impacts all the subject elements as well, due to a hierarchical structure of the tree. This involves the Delete function– deleting the parent element deletes all the children elements.

It is possible to select many elements at once by using a Ctrl or a Shift key the same way as in the Windows Explorer (selecting elements with a Ctrl key pressed allows to select elements one by one, not in a consecutive manner; selecting elements with a Shift key pressed allows to select consecutive elements).

Warning!

To save any modifications in the Workspace Tree hierarchy or to save its naming (not applicable to device tree), the Workspace Tree needs to be saved using a Save or a Save as option available in a File menu or corresponding button on the toolbar.

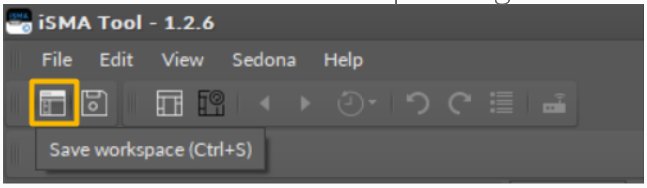


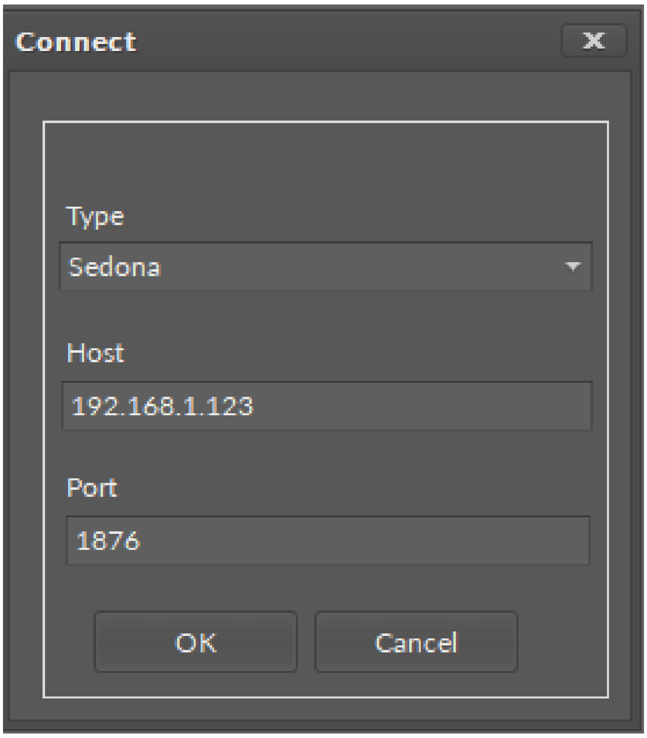
Figure 15. Save Workspace button

5.3.3 Adding and Connecting Device

Adding Device

The iSMA Tool 1.X.X allows to connect only with the controllers from the powered by Sedona Framework iSMA family. Working with other devices than powered by Sedona Framework is blocked (more on this in iSMA Tool Concept).

To connect with the Sedona-based controller, it needs to be added to the Workspace Tree: go to Project > Folder and choose Add Device option from the context menu, which results in adding the device after entering an IP address. First, a proper connection to the controller needs to be established, which requires a proper setting of the IP address of the PC on which the iSMA Tool is active. The figure below shows a window to add the device to the iSMA Tool and a request to enter a proper IP address.


Connect

Type

Sedona

Host

192.168.1.123

Port

1876

Figure 16. Connection dialog window

Worth to Notice:

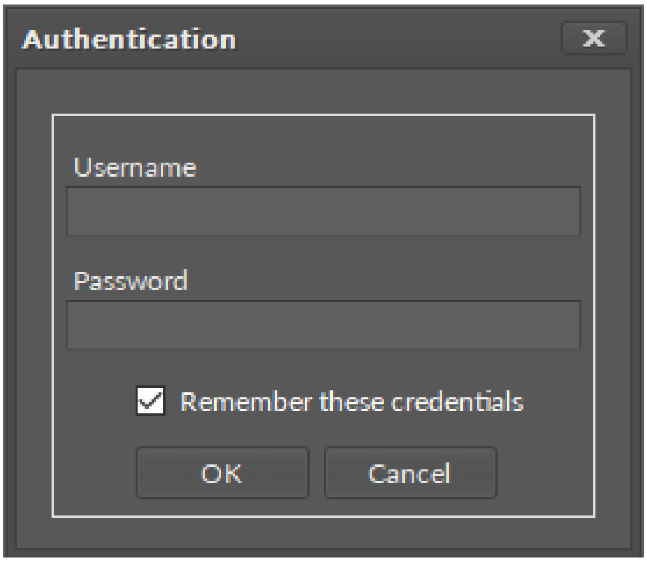
The default IP address for the iSMA-B-AAC20 controller is: 192.168.1.123.

Type field has only one option to choose–Sedona.

Port 1876 is the default communication port for the iSMA-B-AAC20 controller. If the controller has been set to communicate using another port, a proper port number needs to be entered.

Connecting Device

After entering the IP address and pressing Enter or OK, a login window appears, where the username and password need to be entered.



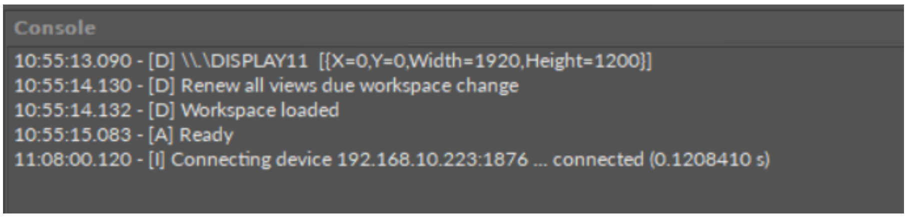
☑ Remember these credentials

Figure 17. Authentication dialog window

> **Warning!**
>
> Saving credentials is dangerous from a security standpoint. Please avoid storing production system passwords.

If the username and password, needed to connect with the controller, are to be remembered, select the Remember these credentials option. The entered data are then saved in the .itw file, which stores the whole structure of the Workspace Tree in a way that prevents access to sensitive data by third parties.

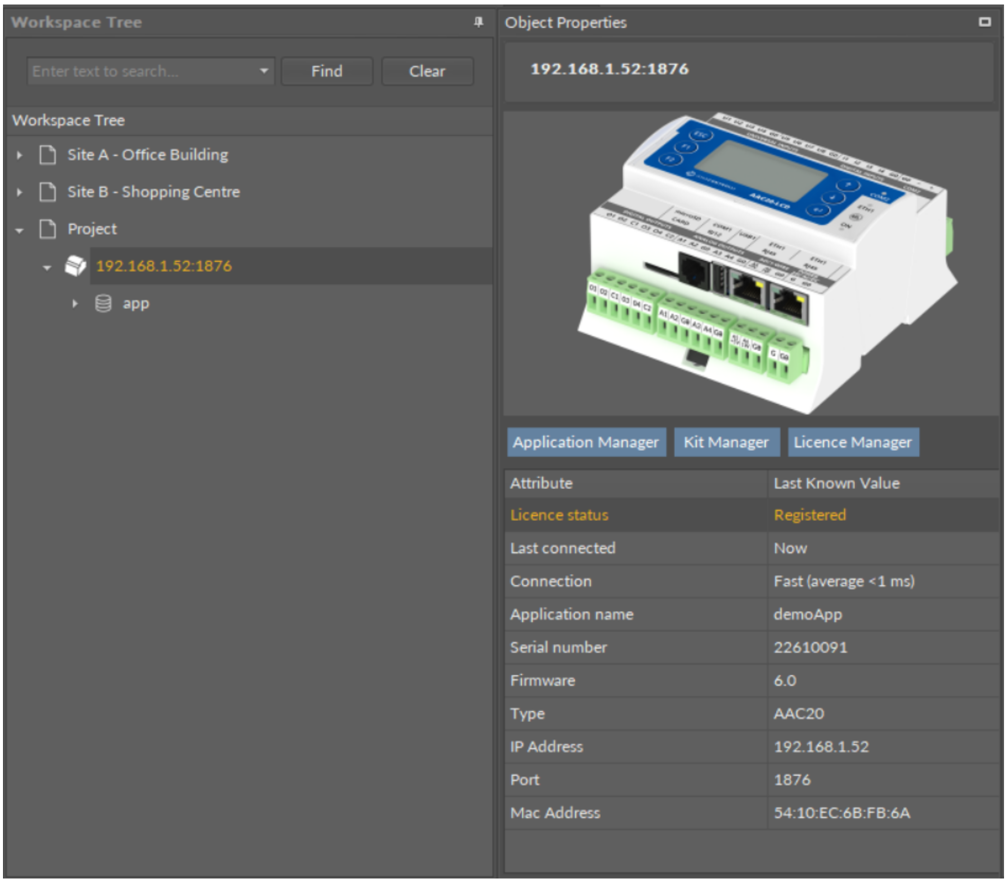
After confirming the entered authorization data by pressing Enter or OK button, the iSMA Tool begins the process of connecting and reading the components tree for the device. The connection process (along with potential issues) is shown in the Console with the log: Connecting device .... connected.



```
10:55:13.090 - [D] \\.\DISPLAY11  [{X=0,Y=0,Width=1920,Height=1200}]
10:55:14.130 - [D] Renew all views due workspace change
10:55:14.132 - [D] Workspace loaded
10:55:15.083 - [A] Ready
11:08:00.120 - [I] Connecting device 192.168.10.223:1876 ... connected (0.1208410 s)
```

Figure 18. Connected device logs in the Console

After a correct connection with the controller, its application tree loads automatically in the Workspace Tree in the app expandable (see the figure below). If the controller is selected in the Workspace Tree view, the Object Properties Window shows its picture and the most important parameters, such as IP address, application name, serial number, or firmware version number. These parameters are available in this window even after disconnecting the controller, which allows offline checking of, e.g., device IP number or its firmware version.



Attribute	Last Known Value
Licence status	Registered
Last connected	Now
Connection	Fast (average <1 ms)
Application name	demoApp
Serial number	22610091
Firmware	6.0
Type	AAC20
IP Address	192.168.1.52
Port	1876
Mac Address	54:10:EC:6B:FB:6A

Figure 19. Connected device in the Object Properties window

Connecting an Added Device

If the iSMA Tool is opened, and all the devices are already added to the project, in order to to connect the device, right-click it, and select the Connect option from the context menu.

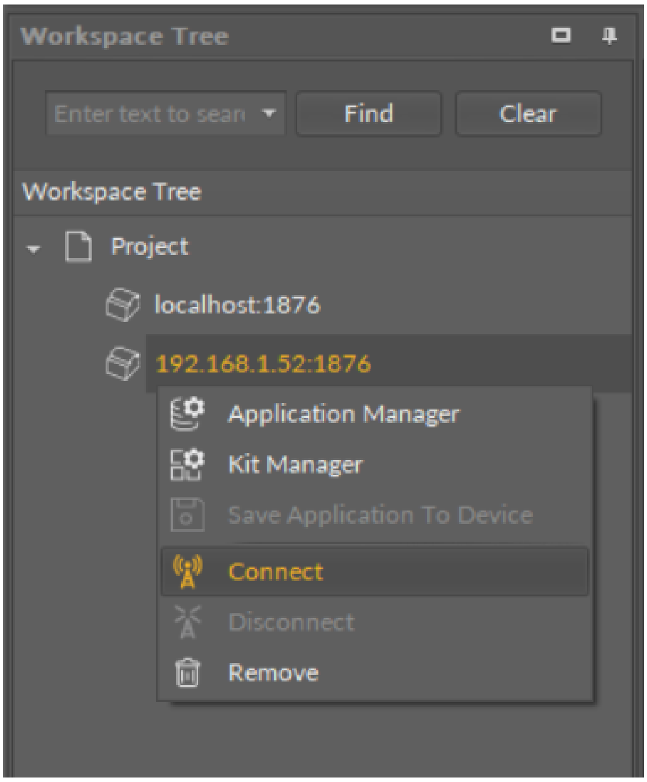
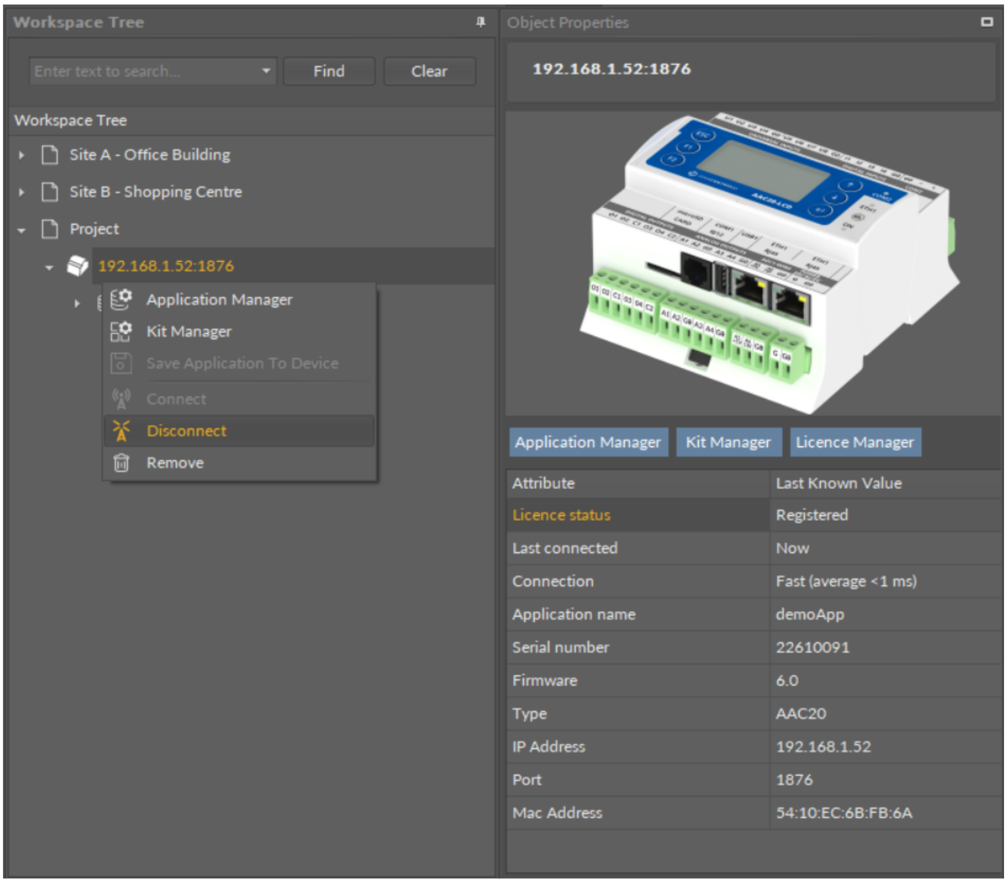


Figure 20. Connecting an added device

Then the Authorization window pops up, either with saved credentials, or ready to fill them in.

5.3.4 Disconnecting Device

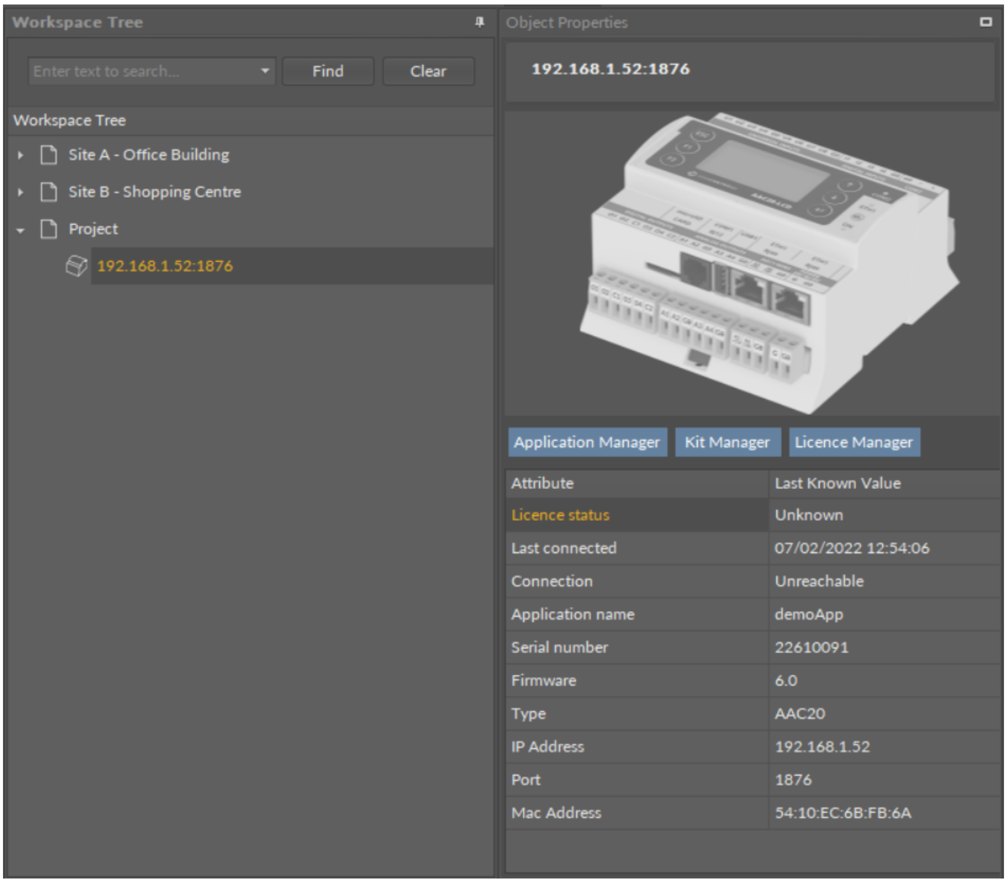
To disconnect the device, select it on the Workspace Tree and choose a Disconnect option from the context menu, then accept.



Attribute	Last Known Value
Licence status	Registered
Last connected	Now
Connection	Fast (average <1 ms)
Application name	demoApp
Serial number	22610091
Firmware	6.0
Type	AAC20
IP Address	192.168.1.52
Port	1876
Mac Address	54:10:EC:6B:FB:6A

Figure 21. Disconnect option in the context menu

After disconnecting the device its icon’s view changes; it is impossible to expand its contents and the figure is grayed out (see the figure below). Double-clicking on the device again begins the reconnection process.



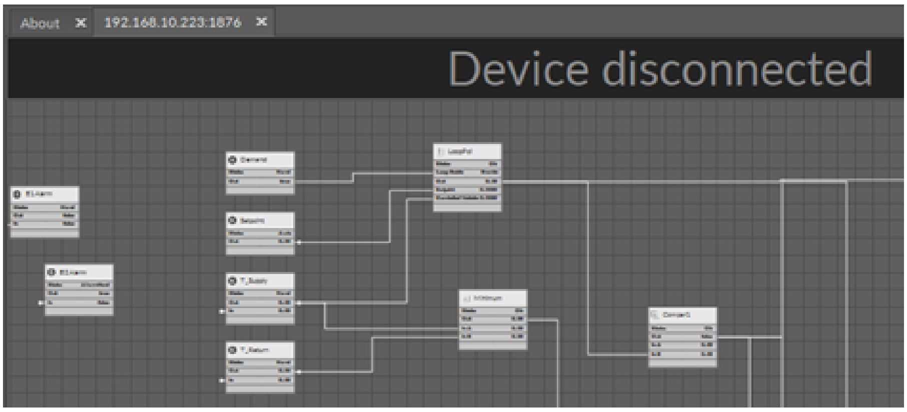
Workspace Tree
- Site A - Office Building
- Site B - Shopping Centre
- Project
  - 192.168.1.52:1876

192.168.1.52:1876

Attribute	Last Known Value
Licence status	Unknown
Last connected	07/02/2022 12:54:06
Connection	Unreachable
Application name	demoApp
Serial number	22610091
Firmware	6.0
Type	AAC20
IP Address	192.168.1.52
Port	1876
Mac Address	54:10:EC:6B:FB:6A

Figure 22. Disconnected device

If the device gets disconnected by mistake, while working with the controller, the program will detect it automatically and inform the user showing a text Device disconnected in the upper part of the view and folding the device tree on the Workspace Tree (see the figure below).



Device disconnected

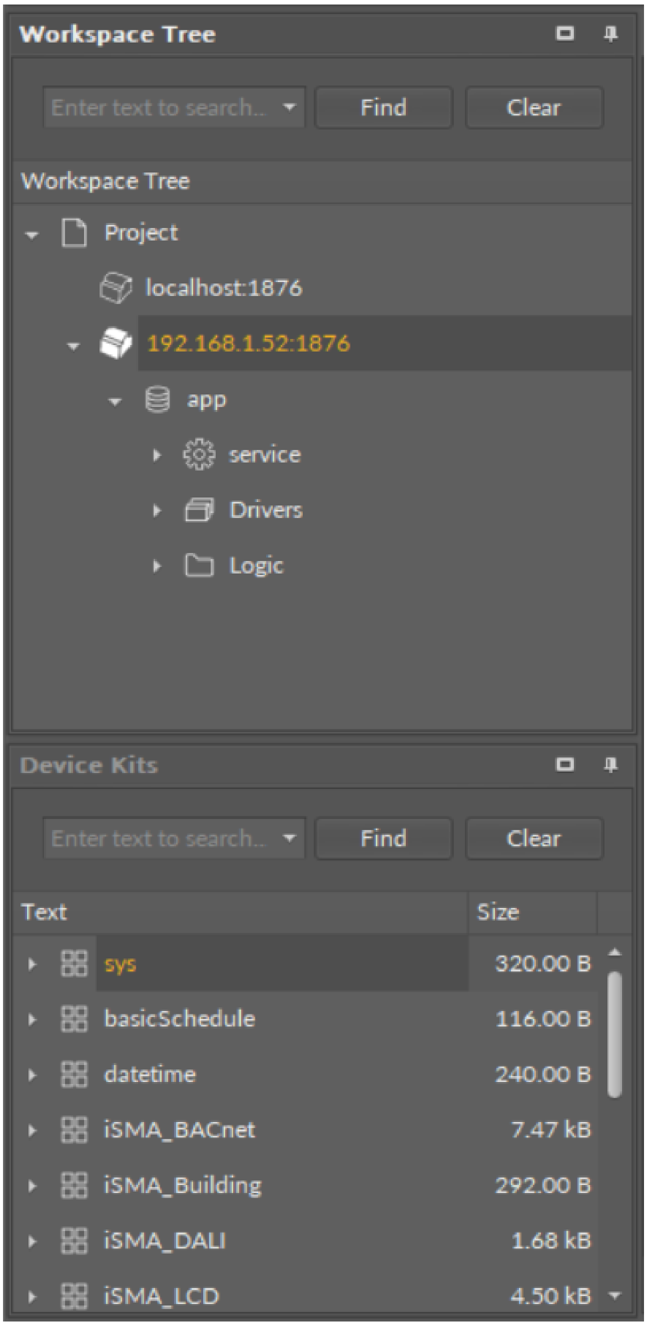
Figure 23. Disconnected device notification

5.3.5 Connected Device View

After connecting with the controller in the Workspace Tree, a subtree appears listing all components available in controller, starting with the app component (application), through service, devices, or Logic (for the iSMA-B-AAC20 controller with default application). Every change made in components, such as editing a component name, its parent, or other parameters in the component device subtree, is saved in the controller. This guarantees, after reloading the data from the controller, the same display of data (components) and synchronization with other users working at the same time on the same controller.

Operations which can be performed on components such as copying, adding, or removing new components, defining connections between components, are described in the further part of this manual.

Immediately after connecting with the device, both the Main screen and Device Kits windows remain unchanged. It is the opening of any component from the device tree that results in showing a specific view in the Main screen and filling the Device Kits window with the proper device kits.



Text	Size
sys	320.00 B
basicSchedule	116.00 B
datetime	240.00 B
iSMA_BACnet	7.47 kB
iSMA_Building	292.00 B
iSMA_DALI	1.68 kB
iSMA_LCD	4.50 kB

Figure 24. The connected device view in the Workspace Tree

5.3.6 Adding and Removing Components

Adding Component

In order to add a new component in the device's application (in the Workspace Tree window), it is necessary to find a specific component in the Device Kits window and drag it to a proper place. The figure below shows the process of dragging a single Counter component from the Device Kits window, as well as a whole group of components, which will be placed under Pumps component.

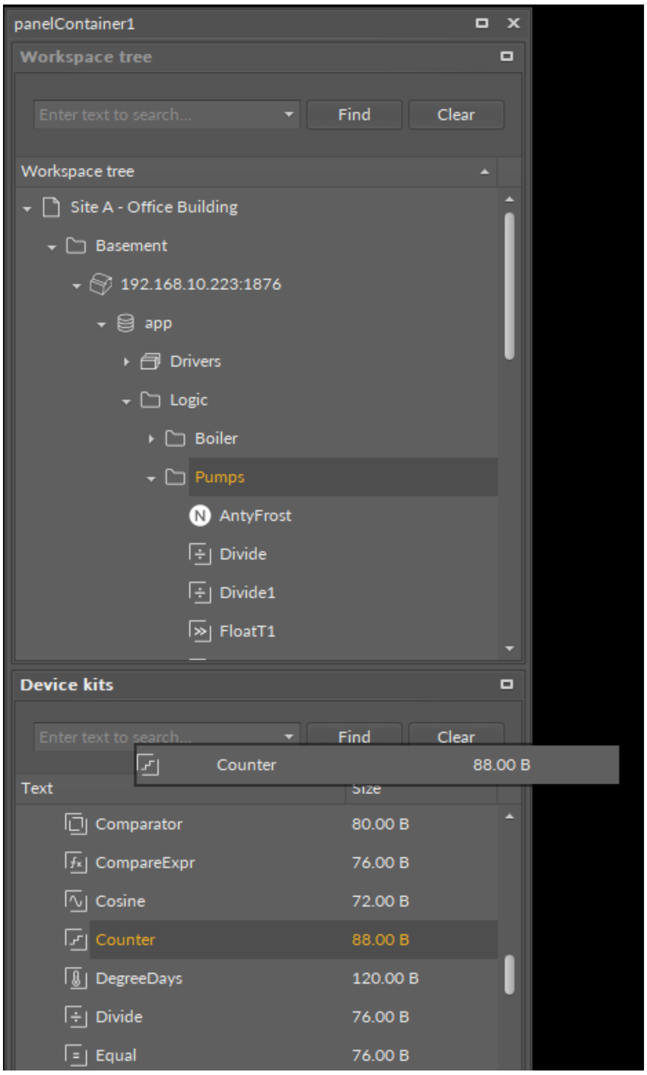
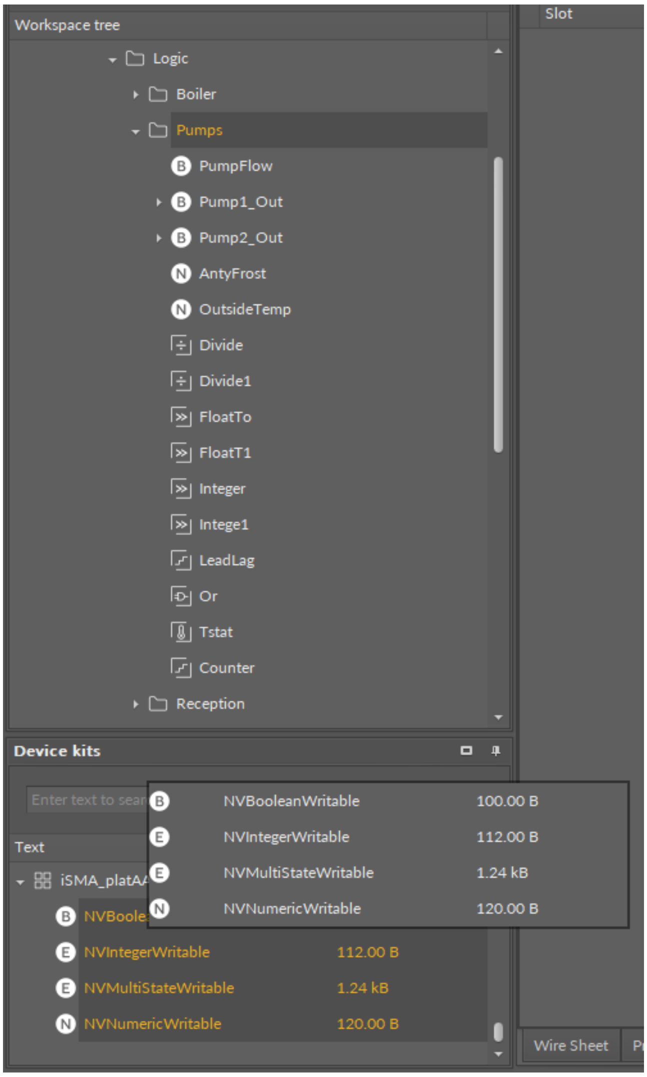


Figure 25. Dragging new components



B	NVBooleanWritable	100.00 B
E	NVIntegerWritable	112.00 B
E	NVMultiStateWritable	1.24 kB
N	NVNumericWritable	120.00 B

E	NVIntegerWritable	112.00 B
E	NVMultiStateWritable	1.24 kB
N	NVNumericWritable	120.00 B

Figure 26. Dragging new components

After dropping a single component in the specific place in the Workspace Tree, a dialog window appears, allowing to name a newly created component. After confirmation of the name, the component is added to the device, and the tree is refreshed.

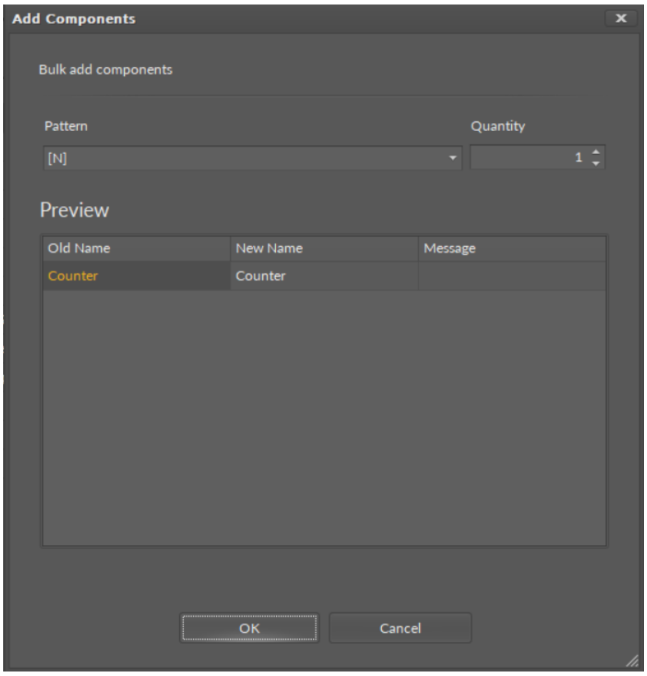


Figure 27. Dialog window for naming new components

Many components can be dragged at once. After dropping them on the Workspace Tree their names will be automatically defined as the component's default name, without the user's participation.

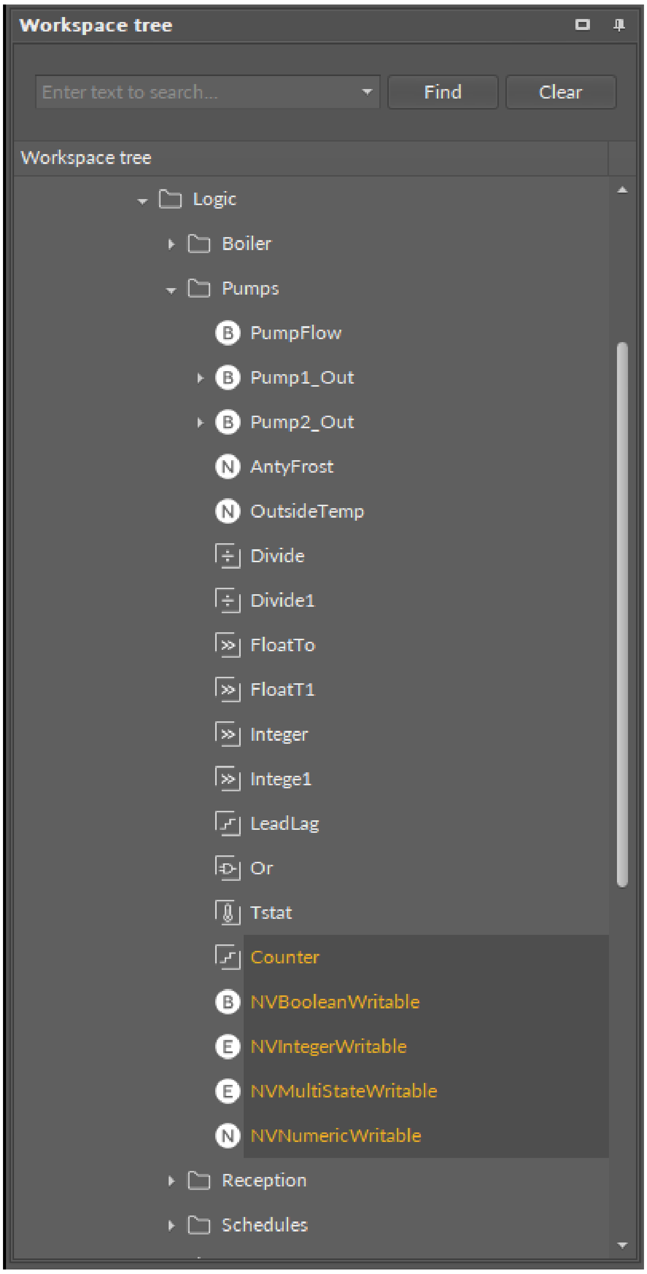


Figure 28. Multiple components added

A second method of adding a component is copying and pasting or duplicating components already existing in the Workspace Tree. In order to do it, select a component in the Workspace Tree window, Property Sheet, or Wire Sheet view, and copy it using a keyboard shortcut Ctrl+C or the context menu using a Copy option. Afterwards, it is essential to choose the component in the Workspace Tree, under which the copied component is pasted with a Ctrl+V operation or the context using a Paste option.

The details of copying/pasting operation as well as duplication are described in the Single Device Functions section.

Removing Component

A component may be removed from the Workspace Tree window level only by selecting it and choosing a Delete option from the context menu. Removal by pressing Delete key on the keyboard is blocked for this window.

5.3.7 Adding and Removing Links

Adding Links

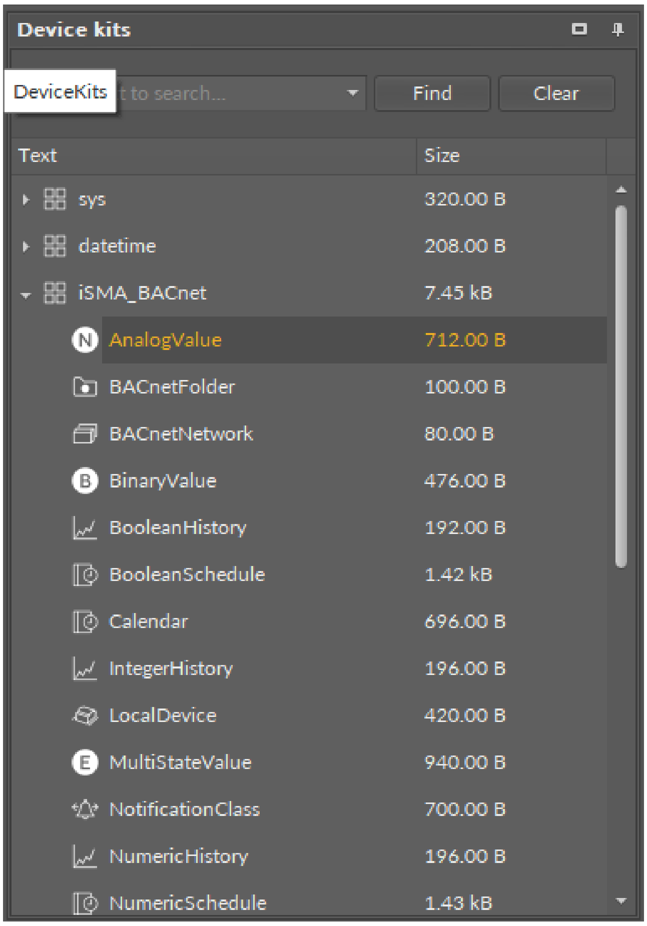
The only way to create a link in the Workspace Tree window is to use the context menu options Link Mark and Link From described in Single Device Functions.

Removing Links

It is not possible to remove a link from the Workspace Tree window directly. To remove the link it is necessary to use the Wire Sheet view.

5.4 Device Kits

The Device Kits is a window with a list of kits available after connecting the iSMA Tool to a device powered by Sedona. This window always shows the kits available in the device connected at the moment. It means that, if there are many devices used in a project, the window contents will be automatically adjusted at the moment of switching from one device to another.



Text	Size
sys	320.00 B
datetime	208.00 B
iSMA_BACnet	7.45 kB
AnalogValue	712.00 B
BACnetFolder	100.00 B
BACnetNetwork	80.00 B
BinaryValue	476.00 B
BooleanHistory	192.00 B
BooleanSchedule	1.42 kB
Calendar	696.00 B
IntegerHistory	196.00 B
LocalDevice	420.00 B
MultiStateValue	940.00 B
NotificationClass	700.00 B
NumericHistory	196.00 B
NumericSchedule	1.43 kB

Figure 29. The Device Kits window

The kits are presented as the main elements of a tree structure in the Device Kits window, and can be expanded to see their components. Each component is presented with its icon, name, and memory it occupies. The icons visible before the component name define it clearly and help to quickly identify the type of component. An icon associated with a particular component is also shown in other views, including the Wire Sheet, Property Sheet, and the path of an editable component. Besides the icons associated to components permanently, there are two special ones, as shown below.

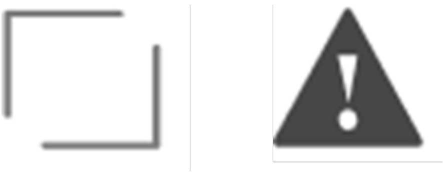


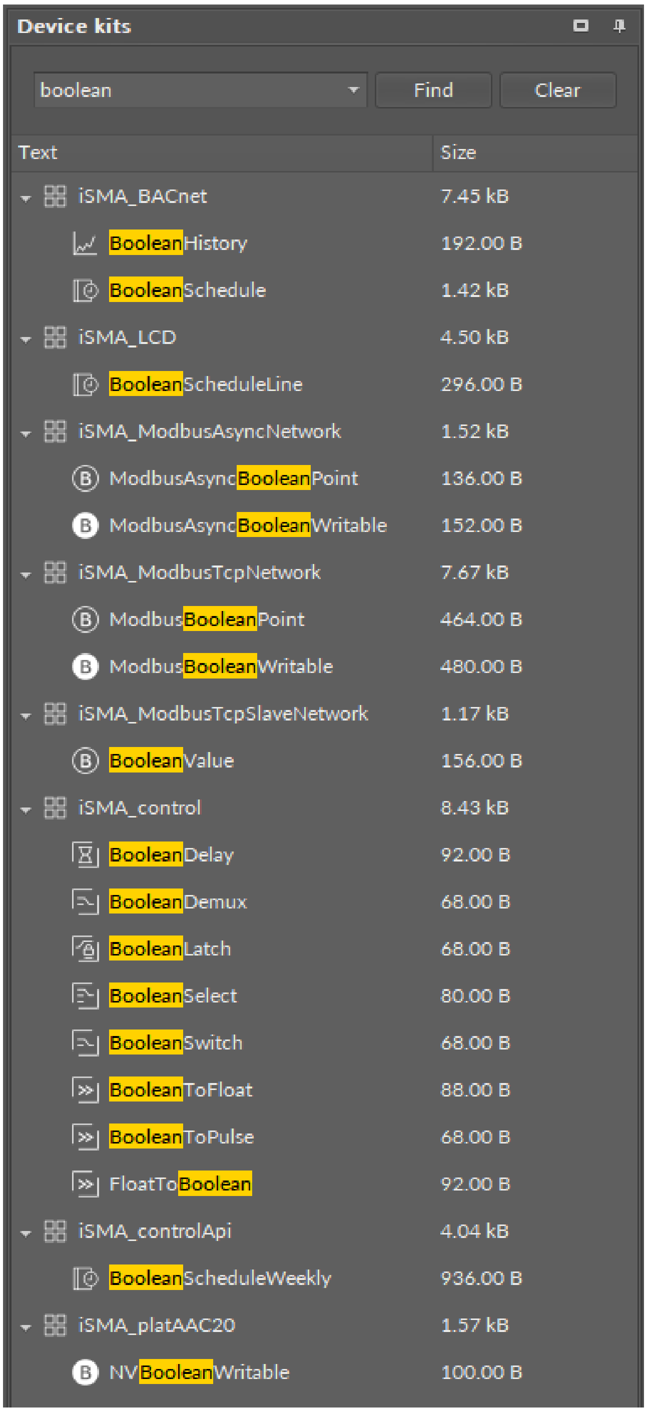
Figure 30. Special icons

The first icon is a default icon for every component without a defined individual icon. The second icon is a warning icon shown for each component, for which the iSMA Tool cannot find a defined individual icon. Such situations may occur while adding custom kits. It requires checking if the path for component icon is correct and if it has been copied to the iSMA Tool icons folder.

5.4.1 Searching the Device Kits

To use a component, it needs to be found and then moved to a required place by drag and drop method, e.g., to the Workspace Tree window.

In order to easily find a specific component among a vast number of components available for a particular controller, it is possible to use a filter available at the top of the window. For this purpose, it is necessary to enter the name or part of the name of the required component into the filter field, and the matching components list will be shown in the Device Kits window. The example below shows a list of components matched with a word Boolean.



Device kits

boolean | Find | Clear

Text	Size
iSMA_BACnet	7.45 kB
BooleanHistory	192.00 B
BooleanSchedule	1.42 kB
iSMA_LCD	4.50 kB
BooleanScheduleLine	296.00 B
iSMA_ModbusAsyncNetwork	1.52 kB
ModbusAsyncBooleanPoint	136.00 B
ModbusAsyncBooleanWritable	152.00 B
iSMA_ModbusTcpNetwork	7.67 kB
ModbusBooleanPoint	464.00 B
ModbusBooleanWritable	480.00 B
iSMA_ModbusTcpSlaveNetwork	1.17 kB
BooleanValue	156.00 B
iSMA_control	8.43 kB
BooleanDelay	92.00 B
BooleanDemux	68.00 B
BooleanLatch	68.00 B
BooleanSelect	80.00 B
BooleanSwitch	68.00 B
BooleanToFloat	88.00 B
BooleanToPulse	68.00 B
FloatToBoolean	92.00 B
iSMA_controlApi	4.04 kB
BooleanScheduleWeekly	936.00 B
iSMA_platAAC20	1.57 kB
NVBooleanWritable	100.00 B

Figure 31. Search results in the Device Kits

It is possible to select and move more than one component simultaneously by using a Ctrl or Shift key.

Using any kit from the Device Kits window means adding an unconfigured component with default parameters. In order to add a component similar to already existing one with pre-configured parameters to the project, it is necessary to use copying or duplicating function on the existing component.

The components dragged to the Wire Sheet view (this view is explained in detail in Wire Sheet) are placed in the location they were dropped. If the components were dropped on a view other than the Wire Sheet view, their physical location (shown on the Wire Sheet view) will be in the upper left corner–point (0,0).

While dragging many components to any view, the iSMA Tool will arrange them in a cascading manner, so they will not overlap.

The components in the Device Kits window cannot be edited and their location cannot be changed.

5.5 Main Screen

The main workspace (Main Screen), where the application is created, is the middle, biggest window of the iSMA Tool. Here, the functional views are available in tabs: the Wire Sheet, Property Sheet, and Slot Sheet.

Navigating between many views in the Main screen is performed by selecting a proper tab placed at the bottom of the window. The figure below shows the Wire Sheet view for component Pumps:

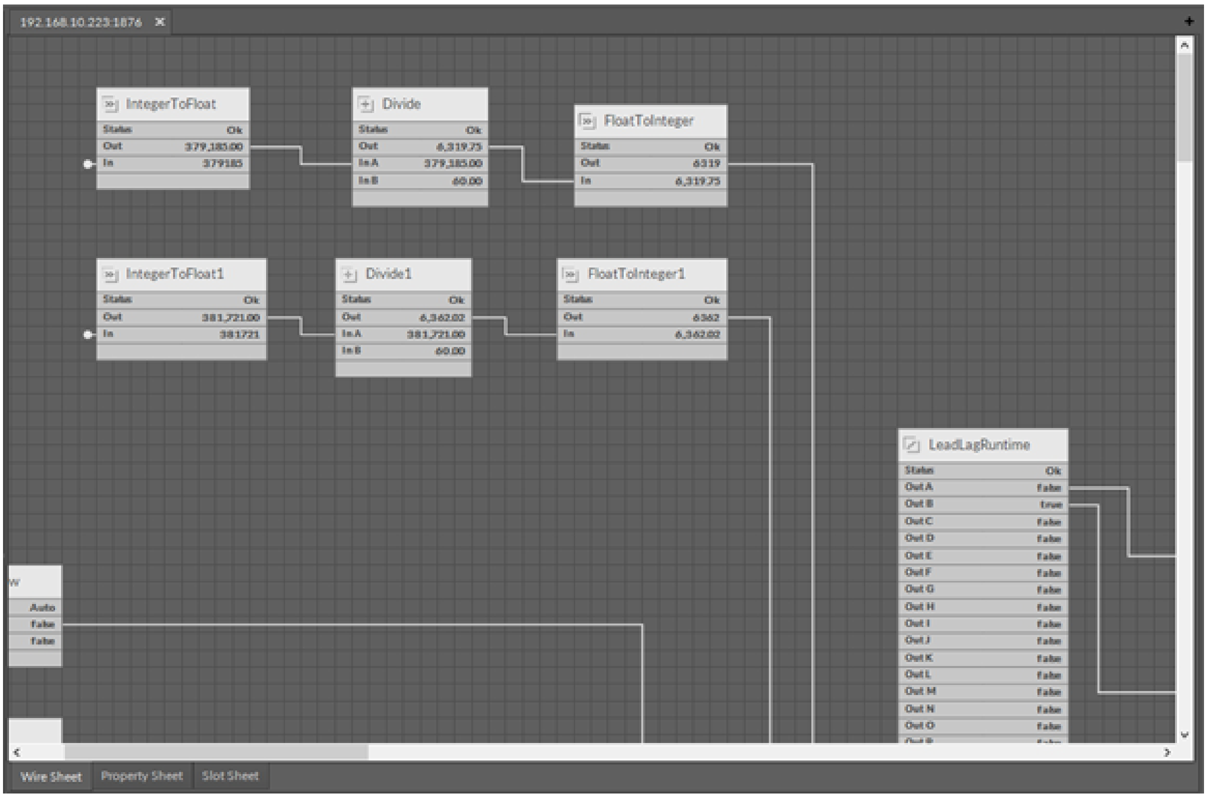


Figure 32. Main screen

To show the component data in the Main screen it is necessary to:

- Double-click the component of interest, which opens in the Main screen the last view used for the selected type of component. The iSMA Tool remembers the preferred view the user was using for every kind of component. For example, the type of the Pumps component (see the figure below) is sys_folder. If the user previously used the Wire Sheet view to see other components of the same type, e.g., the Boiler component, then after double-clicking the Pumps component (associated with the sys_folder) opens in the Wire Sheet view too.
- Open the context menu of a component, hover over the Views option, and choose a required view from the menu (see the figure below). In this case there are three views for the Pumps component, however, for a different component there might be more views available. In this case three basic views are available: the Wire Sheet, Property

Sheet, and Slot Sheet. After opening any of these views, the opened component's type becomes associated with a preferred view (see description above).

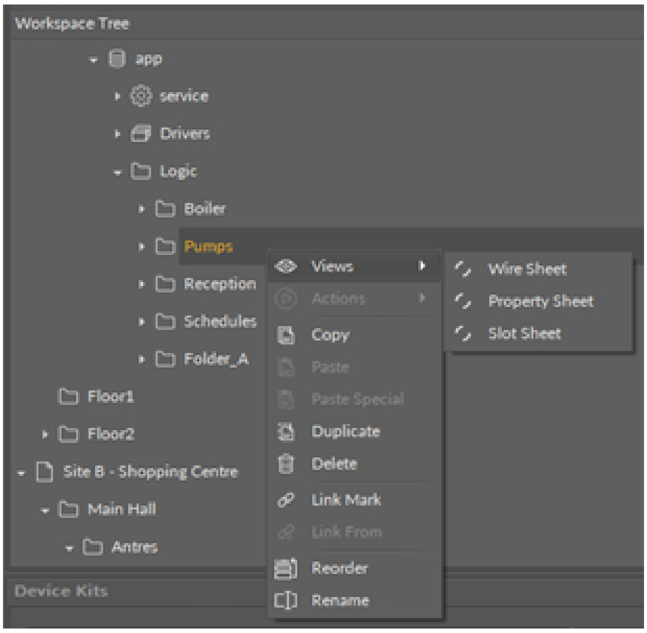


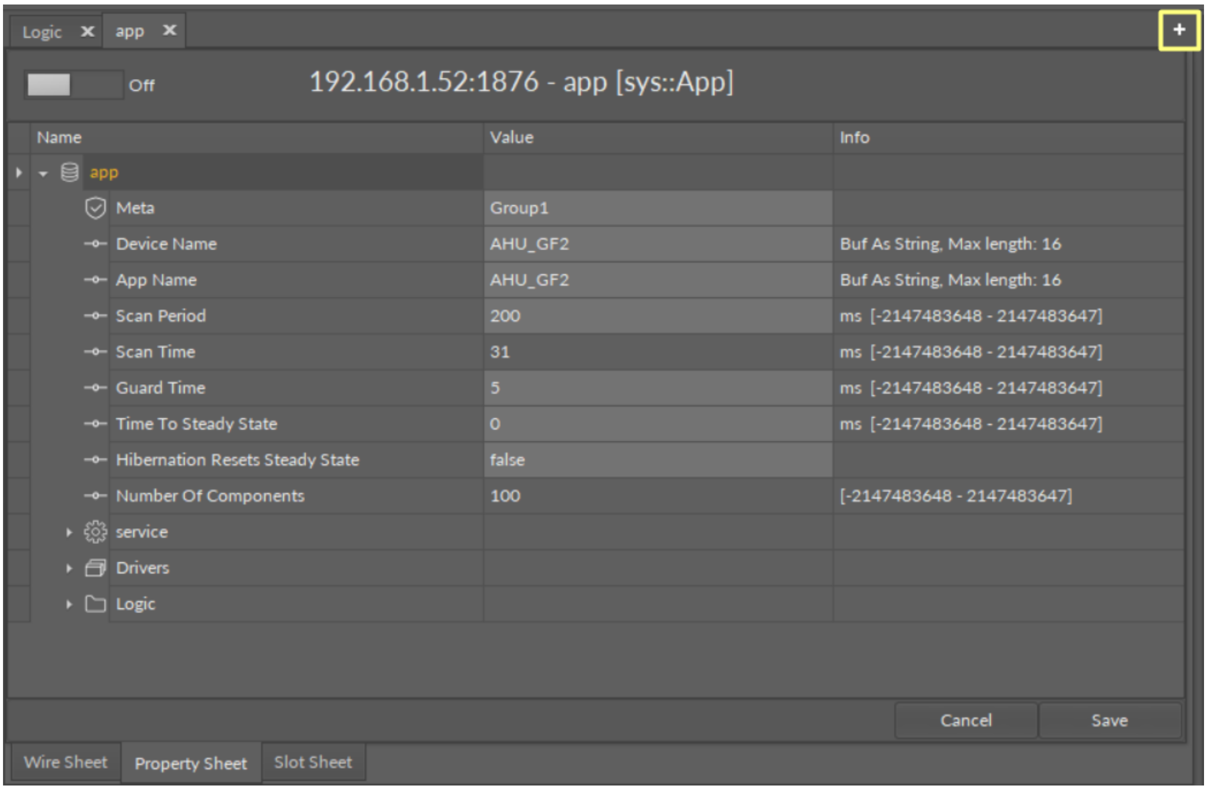
Figure 33. Views options in the context menu

5.5.1 Switching Between Tabs

Once the view is opened in the Main screen, switching between views is possible by selecting a proper tab in the bottom part of the window. The view change is automatically remembered as a view dedicated for a particular component type.

In case one component is opened, choosing another component to edit (using methods described above) replaces all views with data saved earlier for the newly opened component. The default view is the one, which was used recently, or is set by default for the type of component opened.

In order to avoid replacing the data in the tabs, there is a concept of a workspace environment–a new tab, allowing to work on a few components independently, without switching the data in the tabs. To open another, independent tab, use key combination–Ctrl + T, or press icon +, available in the right corner of the tabs bar in the upper part of viewport (see the figure below).



192.168.1.52:1876 - app [sys::App]

Name	Value	Info
app		
Meta	Group1	
Device Name	AHU_GF2	Buf As String, Max length: 16
App Name	AHU_GF2	Buf As String, Max length: 16
Scan Period	200	ms [-2147483648 - 2147483647]
Scan Time	31	ms [-2147483648 - 2147483647]
Guard Time	5	ms [-2147483648 - 2147483647]
Time To Steady State	0	ms [-2147483648 - 2147483647]
Hibernation Resets Steady State	false	
Number Of Components	100	[-2147483648 - 2147483647]
service		
Drivers		
Logic		

Figure 34. Opening independent tab

Tabs raise the work efficiency allowing to quickly switch between views for different components or even devices.

The example below shows that after opening a new tab and choosing the Boiler component to edit, the tab takes the name of the edited component.

Now, the Main screen shows two spaces corresponding to two different components, Pumps and Boiler. These spaces can be switched by choosing a proper tab in the upper part of workspace, and view contents assigned to these workspaces will be shown independently. The figure below shows the Property Sheet view for the second space (second tab), while the first space shows the Wire Sheet view.

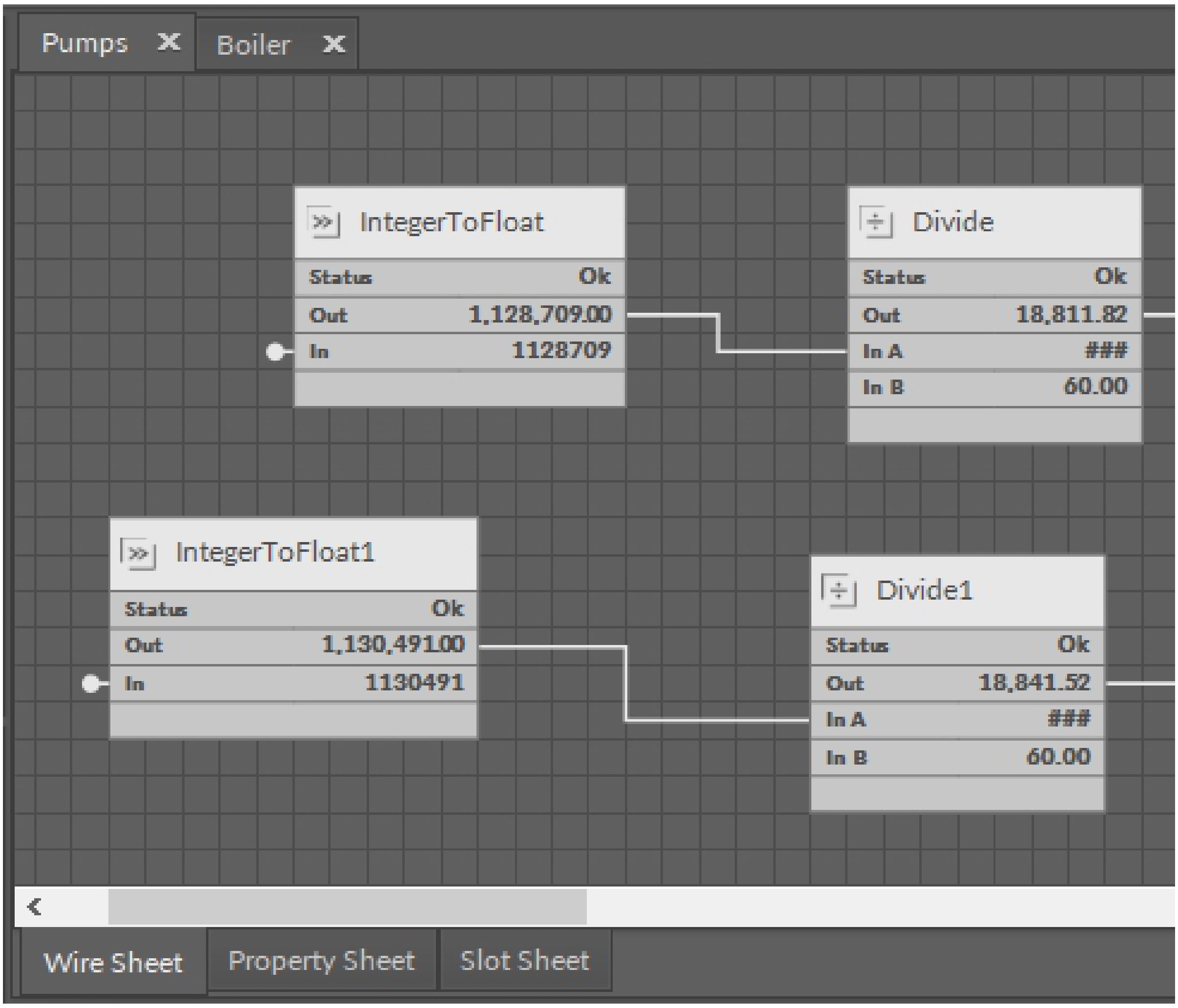
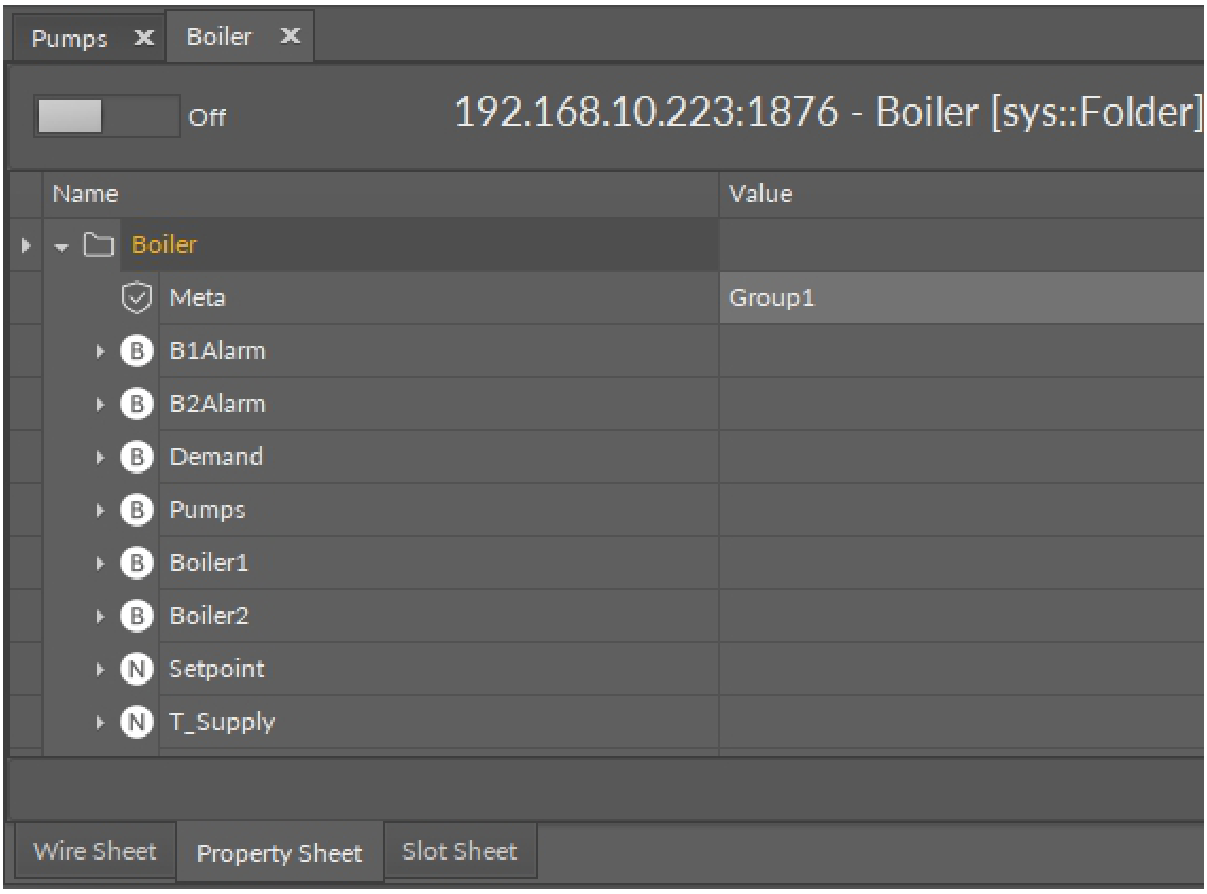


Figure 35. Switching between tabs



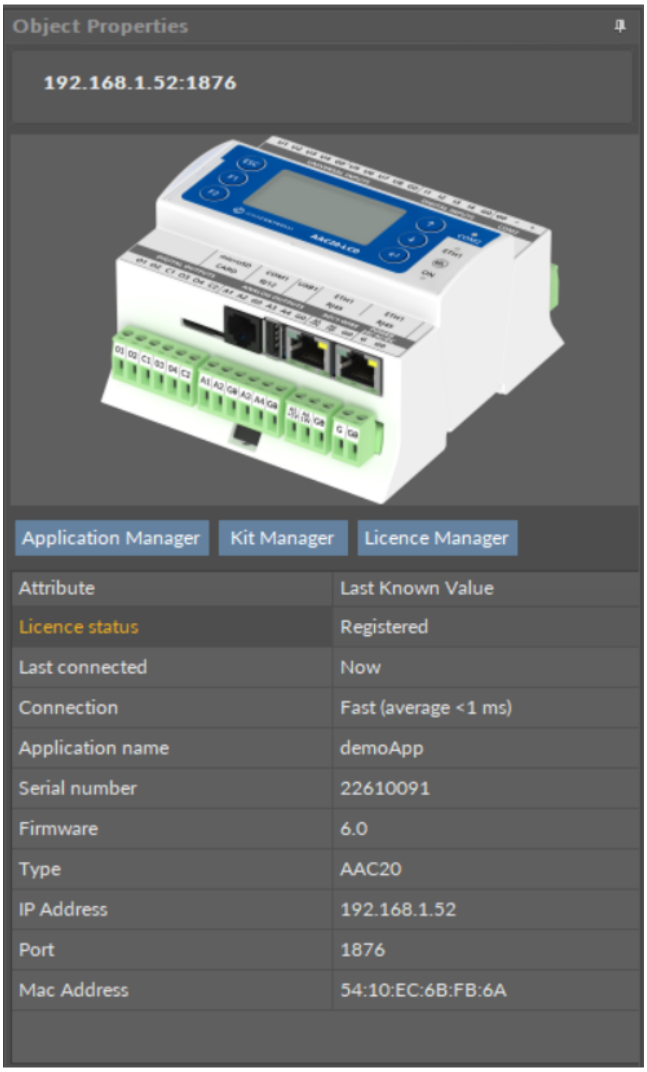
Name	Value
Boiler	
Meta	Group1
B1Alarm	
B2Alarm	
Demand	
Pumps	
Boiler1	
Boiler2	
Setpoint	
T_Supply	

Figure 36. Switching between tabs

5.6 Object Properties

The Object Properties window shows specific information about the selected element, a device, component, or link. It allows a quick preview and editing of a selected element's data without the need for switching, or closing, the current view in the iSMA Tool. The object can be selected in any view and the Object Properties window will show its detailed information.

The view and contents of the Object Properties window may differ depending on the selected element. The look of Object Properties window is described below, depending on the type of the object selected.



Attribute	Last Known Value
Licence status	Registered
Last connected	Now
Connection	Fast (average <1 ms)
Application name	demoApp
Serial number	22610091
Firmware	6.0
Type	AAC20
IP Address	192.168.1.52
Port	1876
Mac Address	54:10:EC:6B:FB:6A

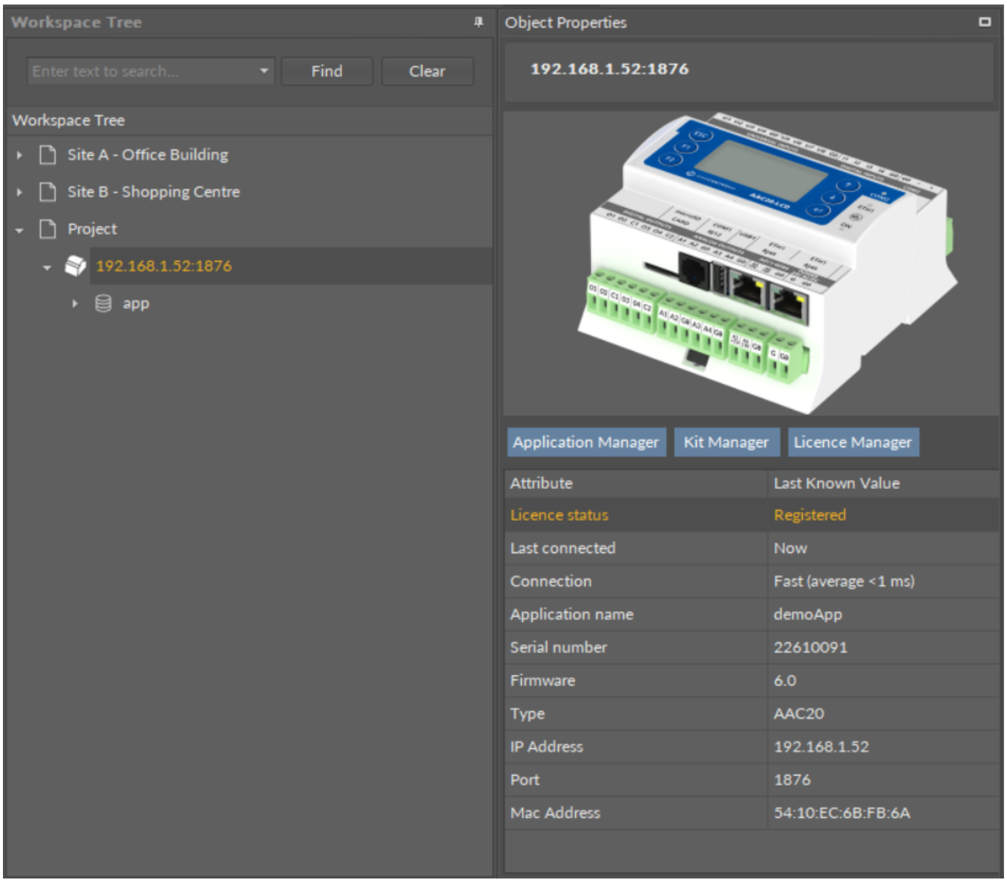
Figure 37. The Object Properties window

5.6.1 Device in Object Properties

After selecting a device in the Workspace Tree window, the Object Properties window presents useful data about the controller:

- license status;
- last connection;
- connection quality;
- application name;
- serial number;
- firmware version;
- device type;
- IP address of the device (available even after changing of device's name–by default the device's name includes its IP number, which is no longer displayed once the name is changed);
- port number;
- MAC address.

Information contained in the Object Properties window is available even after disconnecting the device (in offline mode).



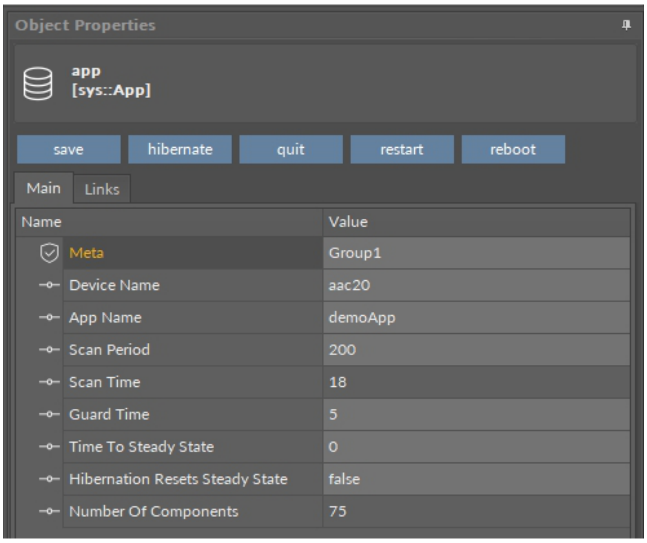
Attribute	Last Known Value
Licence status	Registered
Last connected	Now
Connection	Fast (average <1 ms)
Application name	demoApp
Serial number	22610091
Firmware	6.0
Type	AAC20
IP Address	192.168.1.52
Port	1876
Mac Address	54:10:EC:6B:FB:6A

Figure 38. Selected device in the Object Properties window

5.6.2 Component in Object Properties

After selecting a component, the Object Properties window shows four sections:

- section with a slot list (the Main tab);
- section with a list of outgoing or incoming links to the selected component (the links tab);
- section with a list of configuration data of the selected component (the Info tab)–only available in the Developer mode;
- section with Action buttons (blue buttons above tabs).



Name	Value
Meta	Group1
Device Name	aac20
App Name	demoApp
Scan Period	200
Scan Time	18
Guard Time	5
Time To Steady State	0
Hibernation Resets Steady State	false
Number Of Components	75

Figure 39. The component's view in the Object Properties window

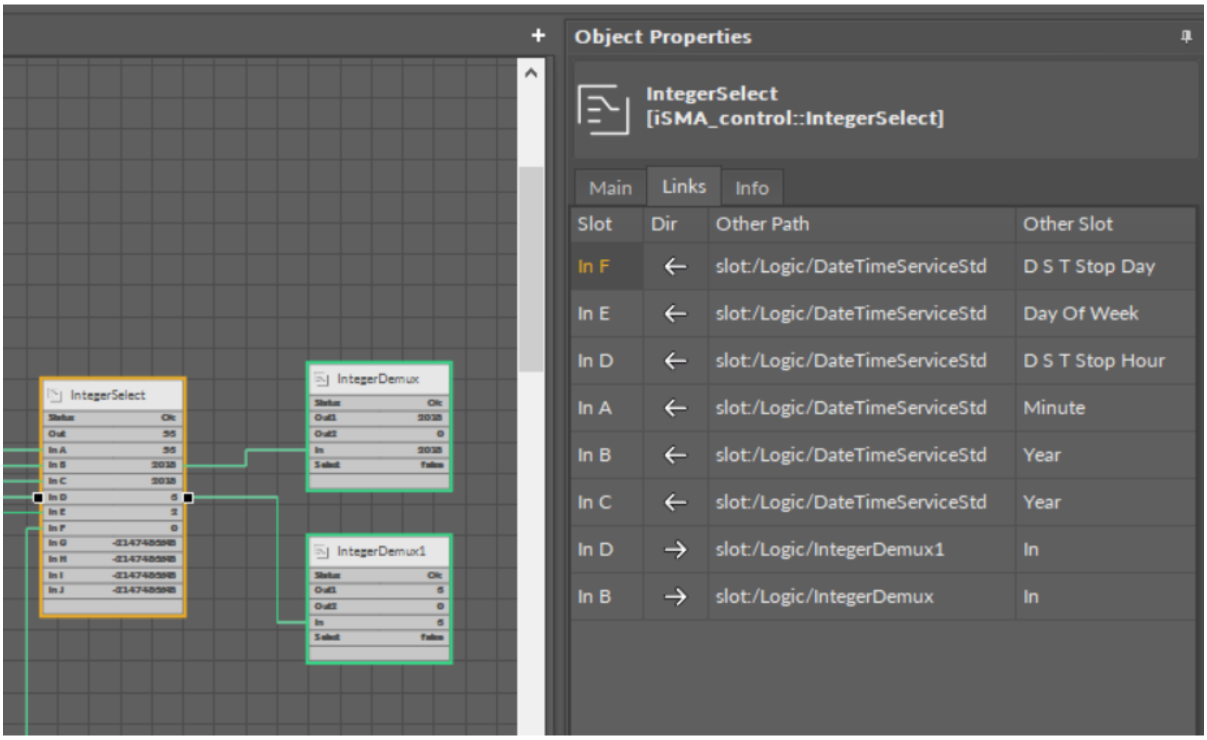
In the upper bar of the window, it shows the icon, name and type of element, which data is displayed. (The figure above shows: app [sys::App]).

- Main tab: allows reading and editing of the selected component’s slots. If the user changes the value of a slot in the Object Properties window, new value is immediately saved in the controller. The Object Properties window works in the auto save mode.

Worth to Notice:

This section is also available to components in kits (in the Device Kits window) and allows to check the slots offered by a particular component even before using it in application.

- Links tab: informs about all incoming and outgoing links with a full location path of target components and the names of target slots. For each link it is possible to choose the following options from the context menu:
    - Go to link: transfers to the Wire Sheet view of a parent of a target component;
    - Go to component: transfers to the Property Sheet view of a target component;
    - Delete link: removes a chosen link.



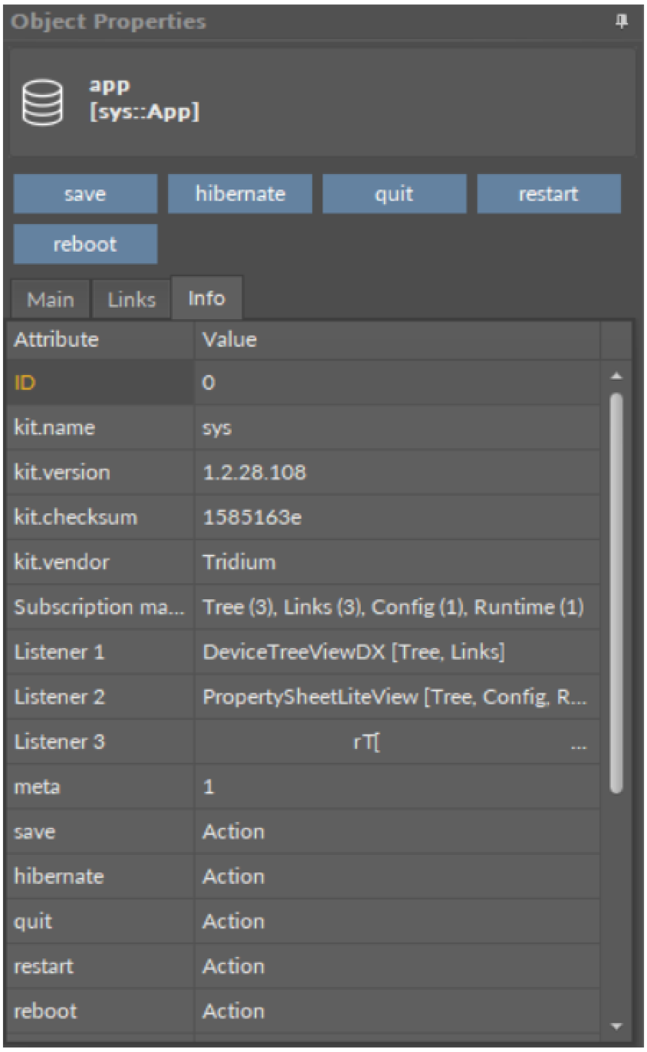
Object Properties

IntegerSelect
[iSMA_control::IntegerSelect]

Slot	Dir	Other Path	Other Slot
In F	←	slot:/Logic/DateTimeServiceStd	D S T Stop Day
In E	←	slot:/Logic/DateTimeServiceStd	Day Of Week
In D	←	slot:/Logic/DateTimeServiceStd	D S T Stop Hour
In A	←	slot:/Logic/DateTimeServiceStd	Minute
In B	←	slot:/Logic/DateTimeServiceStd	Year
In C	←	slot:/Logic/DateTimeServiceStd	Year
In D	→	slot:/Logic/IntegerDemux1	In
In B	→	slot:/Logic/IntegerDemux	In

Figure 40. The Links tab in the Object Properties window

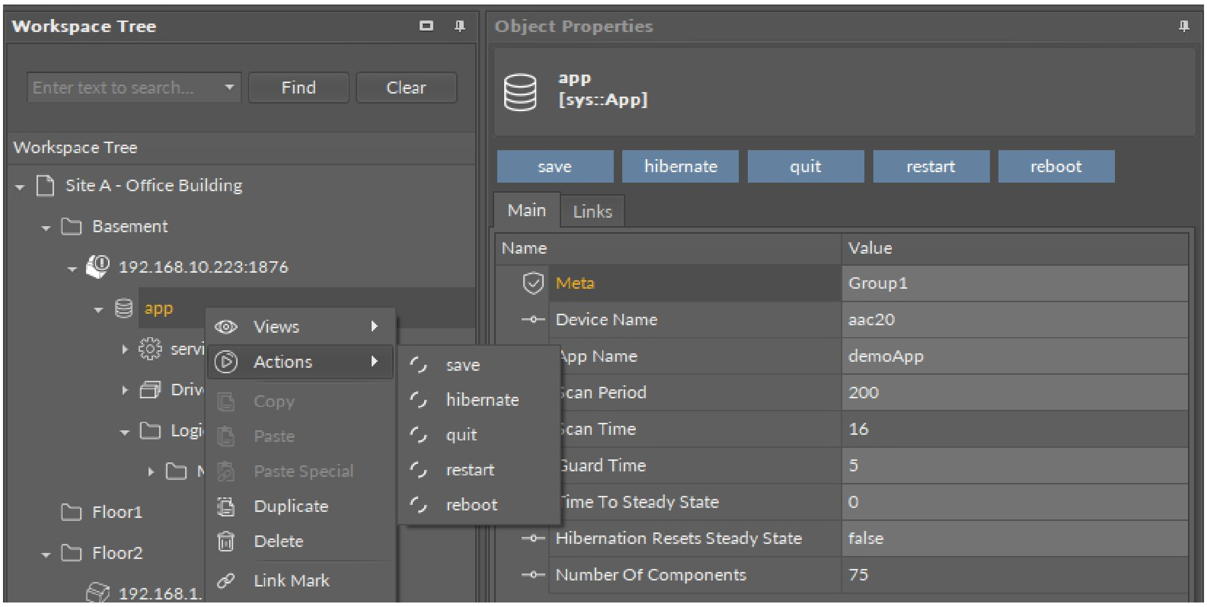
- Info tab: (available only in the Developer Mode) shows the identification and configuration data of the selected component;



Attribute	Value
ID	0
kit.name	sys
kit.version	1.2.28.108
kit.checksum	1585163e
kit.vendor	Tridium
Subscription ma...	Tree (3), Links (3), Config (1), Runtime (1)
Listener 1	DeviceTreeViewDX [Tree, Links]
Listener 2	PropertySheetLiteView [Tree, Config, R...
Listener 3	rT[...
meta	1
save	Action
hibernate	Action
quit	Action
restart	Action
reboot	Action

Figure 41. The Info tab view in the Object Properties window

- Actions section: buttons available in the bar above tabs, allowing a simple execution of Actions on selected component. These are the same Actions which are listed in the component's context menu under the Actions option.



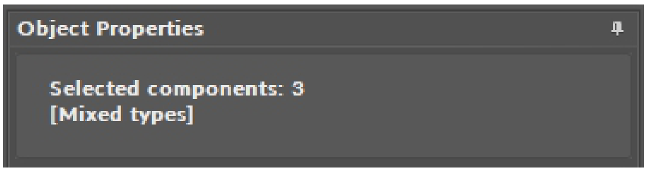
Name	Value
Meta	Group1
Device Name	aac20
App Name	demoApp
Scan Period	200
Scan Time	16
Guard Time	5
Time To Steady State	0
Hibernation Resets Steady State	false
Number Of Components	75

Figure 42. The Actions section in the Object Properties window

Many Components in Object Properties

The iSMA Tool offers the possibility to simultaneously display common properties of many selected components.

Every time the user selects more than one component, the Object Properties Window changes its header to show the number of simultaneously selected components, as shown on the figure below.



Object Properties

Selected components: 3
[Mixed types]

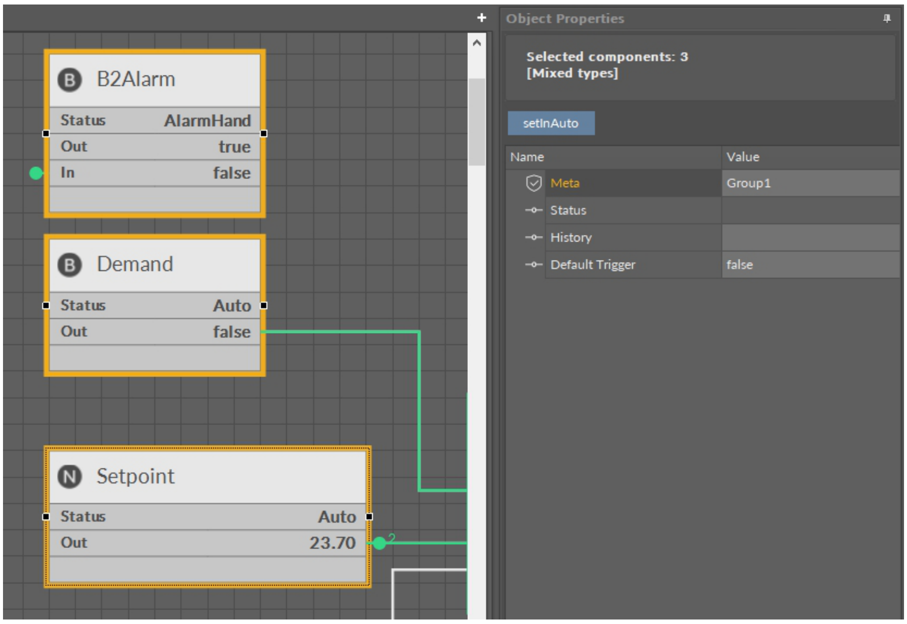
Figure 43. The Obejct Properties header for three selected components of different types

Common Slots

While working with many components simultaneously, the Object Properties window changes its work mode to show only the slots common for all selected components. The value of the slot shown in this mixed view is only displayed if all selected components have exactly the same value for this slot; otherwise, the value field is empty.

Regardless of whether the shown slot displays a value or not, if it is an editable slot, entering a new value overwrites it in all selected components. If this process is not successful, the Console will show an appropriate information.

As shown on the figure below, selecting 3 components of different types (NVBooleanWritable and NVNumericWritable) resulted in showing only 4 common slots in the Object Properties window. 'Meta' and 'Default Trigger' slots have identical values in all components and they are displayed in the mixed view. The remaining 2 slots, Status and History, have different values, therefore there are empty spaces in the place of value fields.



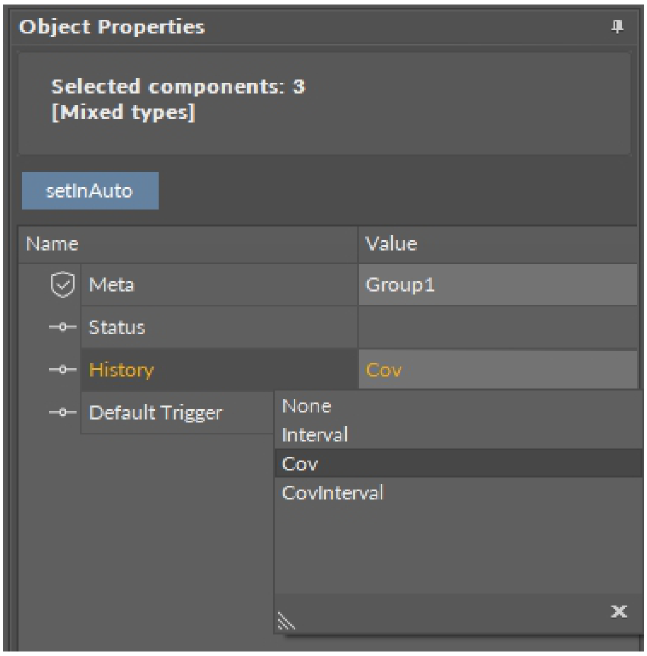
Object Properties

Selected components: 3
[Mixed types]

Name	Value
Meta	Group1
Status	
History	
Default Trigger	false

Figure 44. Common slots

Using this multi-select function for an editable slot enables entering a new value (e.g., Cov for the History slot) which will be sent to all 3 components at once, as is shown on the figure below.


Object Properties

Selected components: 3
[Mixed types]

Name	Value
Meta	Group1
Status	
History	Cov
Default Trigger	

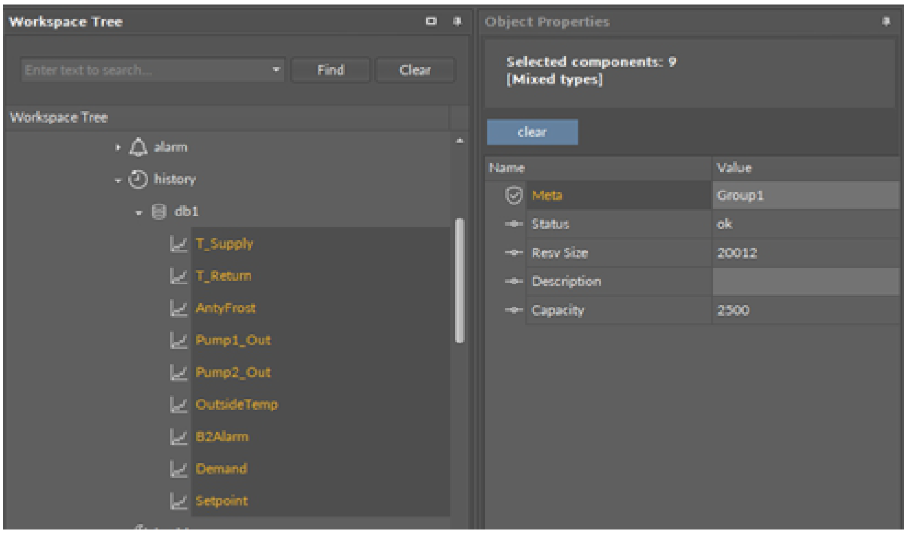
Figure 45. Changing values in common slots

Setting the same values for various components simultaneously, as described above, allows setting common setpoints or configurations for many components at once, e.g., Interval time for historical ones, Enable to activate network variables, etc.

Common Actions

The case of common actions is similar to setting a new value for common slots. Assuming the selected components share the same actions, then the buttons of these actions appear in the Object Properties window. Pressing the action button will execute the action in all selected components.

A good example is an action erasing saved historical samples in all historical components. For this purpose all historical components in the Workspace Tree have been selected and the Object Properties window has shown modified content, as shown on the figure below:

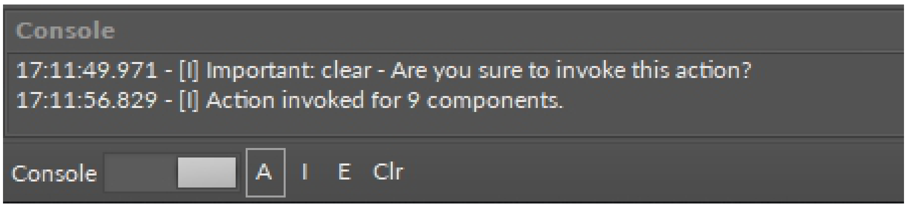


Selected components: 9
[Mixed types]

Name	Value
Meta	Group1
Status	ok
Resv Size	20012
Description	
Capacity	2500

Figure 46. Common actions

Pressing the Clear action button and confirming the action in a pop-up window results in erasing the history for all variables, which also appears as a message in the Console.



```
17:11:49.971 - [I] Important: clear - Are you sure to invoke this action?
17:11:56.829 - [I] Action invoked for 9 components.
```

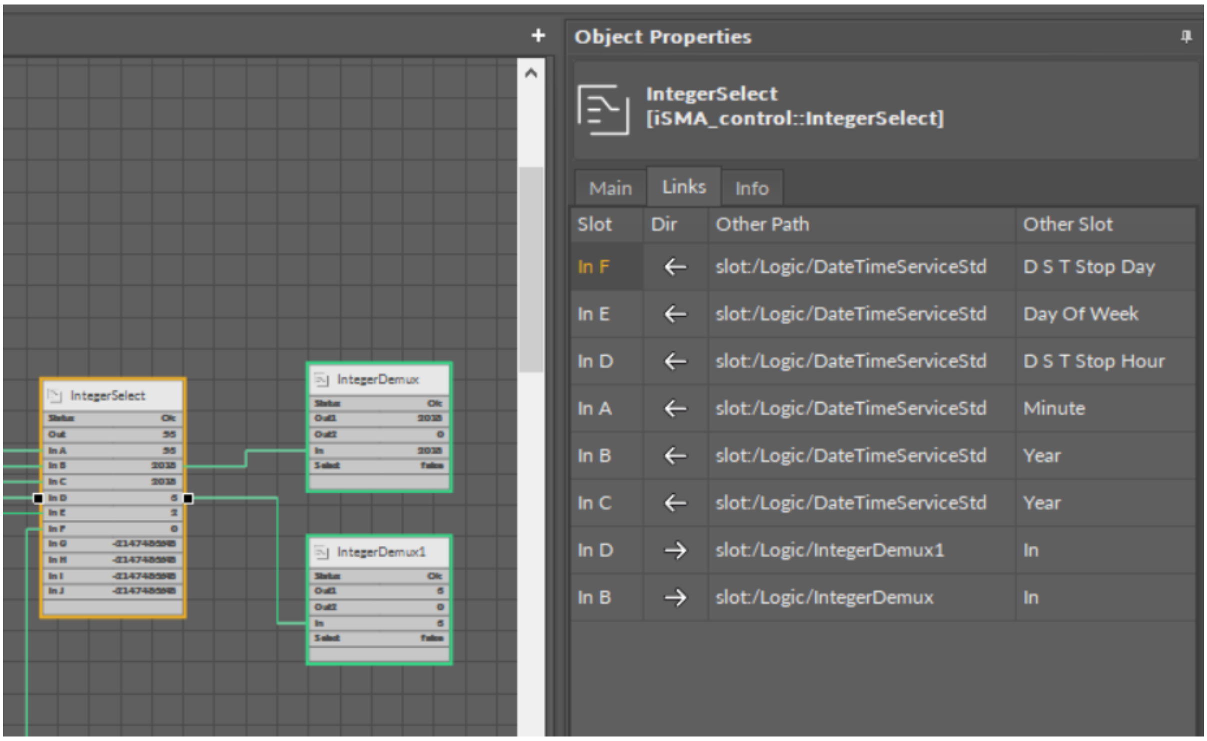
Figure 47. Console log for a common action

5.6.3 Links Tab

A Links Tab view is a view showing a list of all links incoming and outgoing of the selected component in Object Properties view. The link's full location path is displayed in the Links tab, including the name of the target slot.

This view is a dynamic one, and it is updated constantly by downloading data directly from the controller.

The figure below shows the component selected in the Wire Sheet view. The Object Properties window has instantly shown all its links in the Links tab.



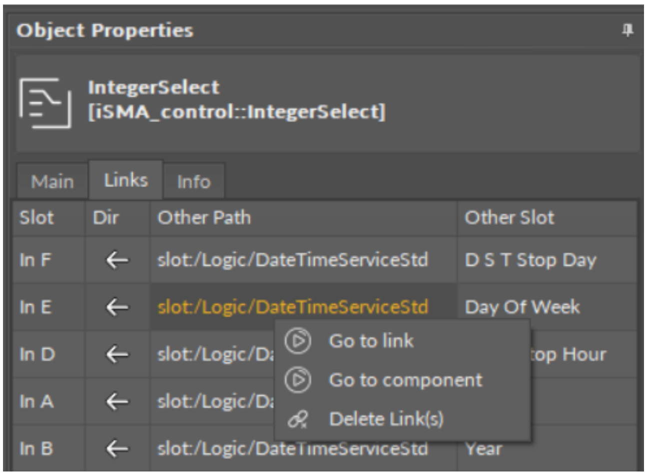
IntegerSelect
[iSMA_control::IntegerSelect]

Slot	Dir	Other Path	Other Slot
In F	←	slot:/Logic/DateTimeServiceStd	D S T Stop Day
In E	←	slot:/Logic/DateTimeServiceStd	Day Of Week
In D	←	slot:/Logic/DateTimeServiceStd	D S T Stop Hour
In A	←	slot:/Logic/DateTimeServiceStd	Minute
In B	←	slot:/Logic/DateTimeServiceStd	Year
In C	←	slot:/Logic/DateTimeServiceStd	Year
In D	→	slot:/Logic/IntegerDemux1	In
In B	→	slot:/Logic/IntegerDemux	In

Figure 48. The Links tab

Each link in the Link tab has the context menu available with the following options:

- Go to link: go to the Wire Sheet tab of the parent of the target component;
- Go to component: go to the Property Sheet tab of the target component;
- Delete link: remove a chosen link.

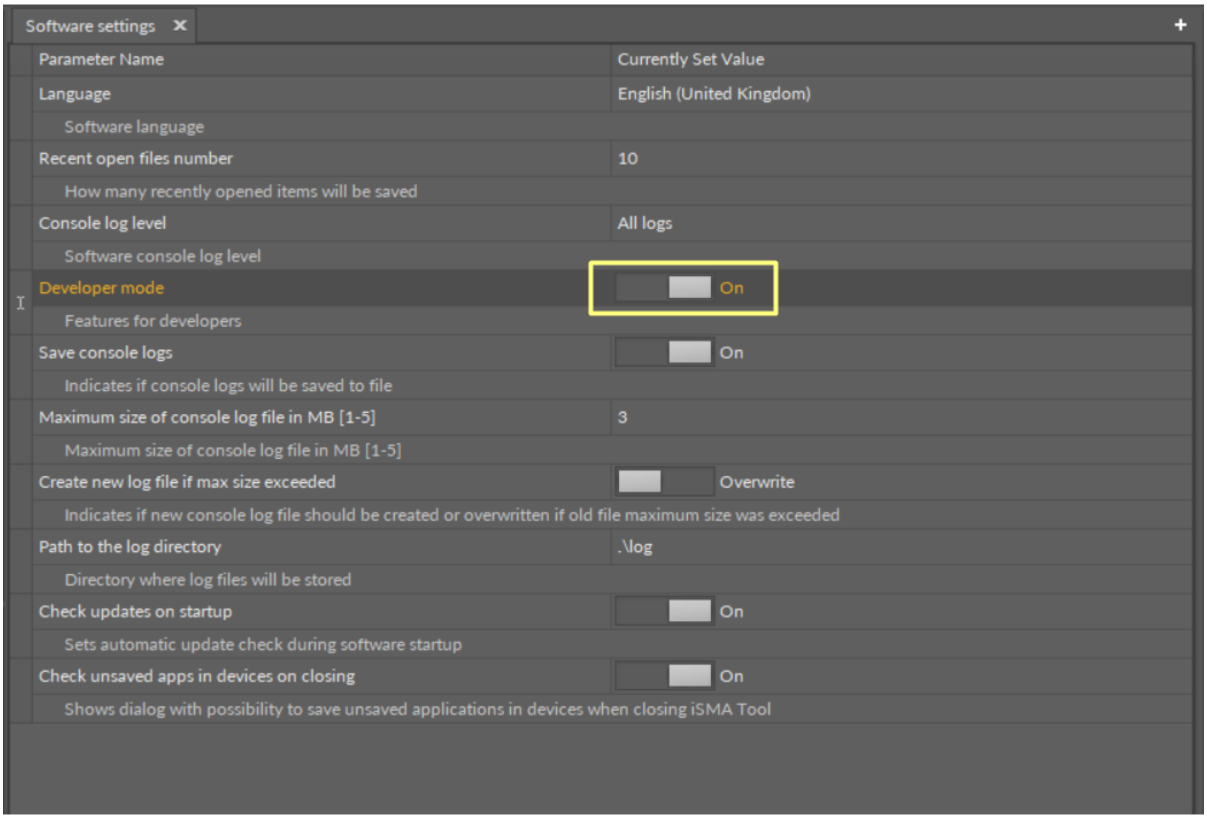


Slot	Dir	Other Path	Other Slot
In F	←	slot:/Logic/DateTimeServiceStd	D S T Stop Day
In E	←	slot:/Logic/DateTimeServiceStd	Day Of Week
In D	←	slot:/Logic/D:	:op Hour
In A	←	slot:/Logic/D:	
In B	←	slot:/Logic/DateTimeServiceStd	Year

Figure 49. A link's context menu

5.6.4 Object Properties in Developer Mode

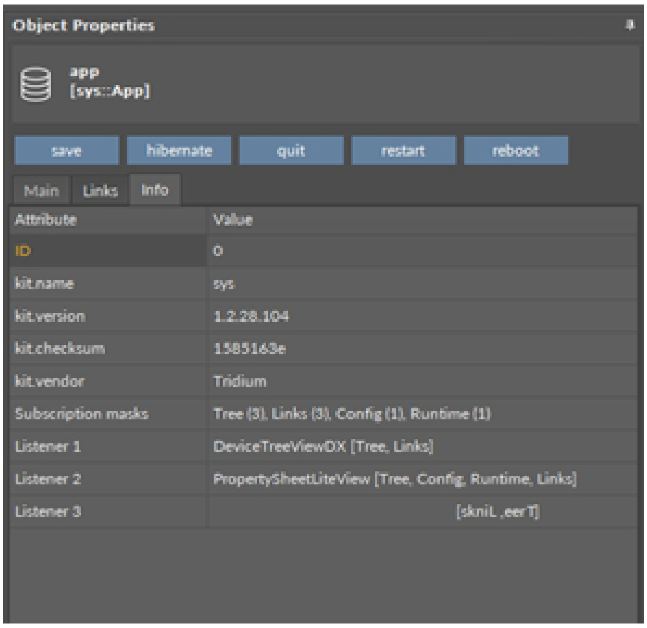
The Object Properties window may provide advanced data such as a component ID, version of the component's library, and other technical data useful to an advanced user. These data are available in the Info tab, which appears in the Object Properties window after switching on the Developer Mode in the iSMA Tool settings. To switch on the Developer Mode, go to the main menu File > Settings; the Developer Mode is switched on by toggling in the Settings window.



Parameter Name	Currently Set Value
Language	English (United Kingdom)
Software language	
Recent open files number	10
How many recently opened items will be saved	
Console log level	All logs
Software console log level	
Developer mode	On
Features for developers	
Save console logs	On
Indicates if console logs will be saved to file	
Maximum size of console log file in MB [1-5]	3
Maximum size of console log file in MB [1-5]	
Create new log file if max size exceeded	Overwrite
Indicates if new console log file should be created or overwritten if old file maximum size was exceeded	
Path to the log directory	.\log
Directory where log files will be stored	
Check updates on startup	On
Sets automatic update check during software startup	
Check unsaved apps in devices on closing	On
Shows dialog with possibility to save unsaved applications in devices when closing iSMA Tool	

Figure 50. Turning on the Developer Mode

The figure below shows the component app with advanced data in the Info tab.



Attribute	Value
ID	0
kit.name	sys
kit.version	1.2.28.104
kit.checksum	1585163e
kit.vendor	Tridium
Subscription masks	Tree (3), Links (3), Config (1), Runtime (1)
Listener 1	DeviceTreeViewDX [Tree, Links]
Listener 2	PropertySheetLiteView [Tree, Config, Runtime, Links]
Listener 3	[skniL ,eerT]

Figure 51. The Info tab in the Object Properties window

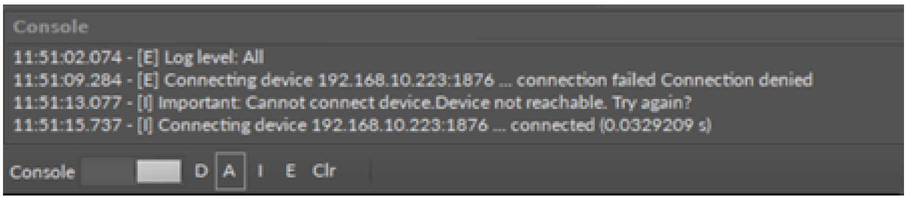
Advanced data in the Info tab are also available after selecting a component in the Device Kits window, which enables checking who the author of particular kit is and what its version is.

5.7 Console and Logs

The Console is a window where the iSMA Tool informs the user about the application status and its operations. While performing user's tasks or processes such as object removal or establishing connection with a controller, the Console window shows the logs of what is being performed at the moment.

The basic concept in the iSMA Tool is that the Console shows all detailed messages, while only some of them are additionally displayed as a pop-up window.

It is possible to filter the log contents in the Console by using four buttons: D, A, I and E (see the figure below).



```
11:51:02.074 - [E] Log level: All
11:51:09.284 - [E] Connecting device 192.168.10.223:1876 ... connection failed Connection denied
11:51:13.077 - [I] Important: Cannot connect device.Device not reachable. Try again?
11:51:15.737 - [I] Connecting device 192.168.10.223:1876 ... connected (0.0329209 s)
```

Figure 52. The Console view

Choosing letters from D to E the number of logs shown in the Console window is successively narrowed. The particular letters meaning is as follows:

D: Debug: this mode shows the biggest amount of data in the Console. This setting is recommended for programming of controllers for diagnostic purposes by the advanced iSMA Tool users. If this option is not available, it needs to be activated manually in the iSMA Tool settings (see the description further in this chapter);

- A: All: shows all data; this is the basic recommended setting;
- I: Important: shows only the most important data;
- E: Errors: the Console shows only error messages; this is the most restrictive filter;

- Clr: Clear: clears all the contents in the Console;
- Turn the Console on/off: freezes the Console and stops showing new events in the Console.

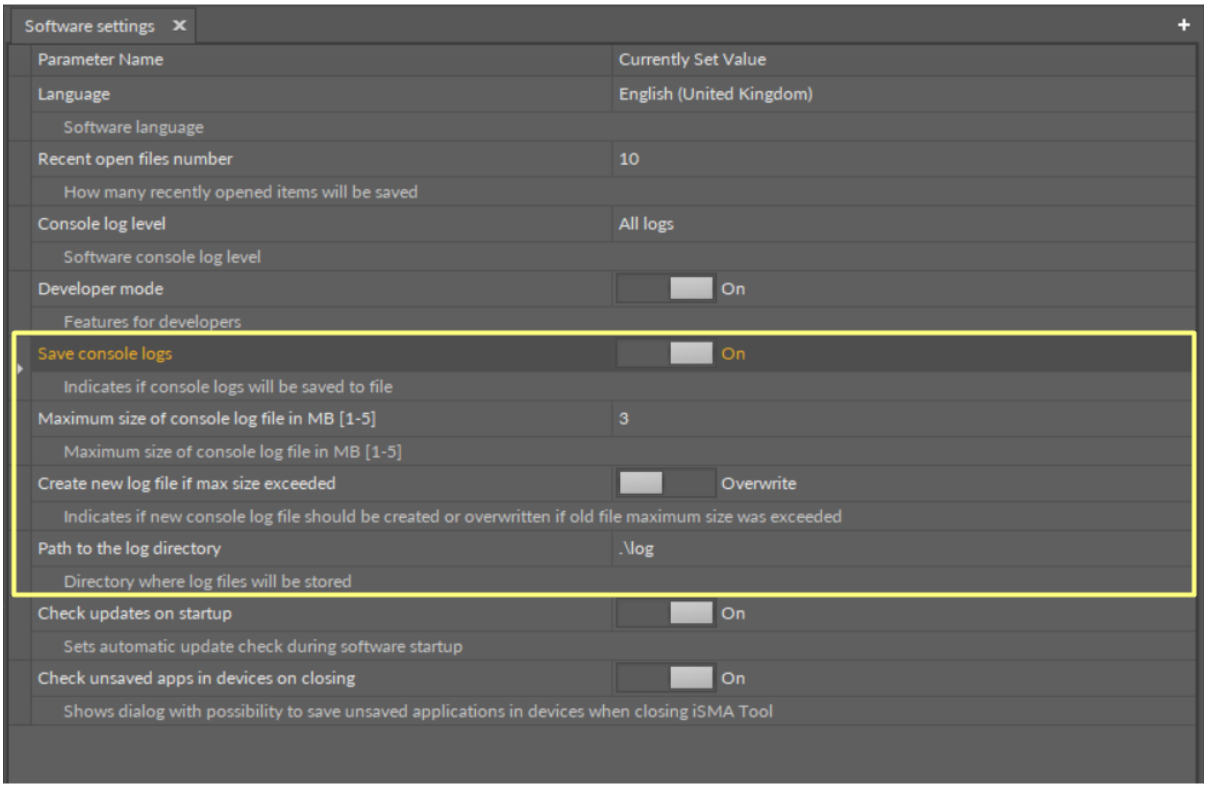
Regardless of the chosen filter, very important errors are always shown in the Console, except when the Console is turned off.

The Console window contents including all filtered information not shown in the Console are automatically saved in the log file to store the information for a longer period. It is therefore possible to open the log file for a later analysis using any text editor. The log files are by default saved in the log folder. Both the name of the folder for saving logs and its location may be changed by defining a path in the Settings (see the description further in this chapter).

The iSMA Tool allows to define the size of a single log file, as well as the course of conduct after exceeding the maximum log file size. The file size may be from 1 MB to 5 MB (by default, it is 3 MB) and there are two possible courses of conduct after exceeding the maximum log file size:

- creating a new log file and keeping previous files;
- complete wiping of the actual log file and start recording again.

The above settings can be modified in the Software settings in the main menu File -> Settings. The Software settings allow to individually set various parameters of the iSMA Tool. Among all, there are those responsible for log configuration and the Console window performance.



Parameter Name	Currently Set Value
Language	English (United Kingdom)
Software language	
Recent open files number	10
How many recently opened items will be saved	
Console log level	All logs
Software console log level	
Developer mode	On
Features for developers	
Save console logs	On
Indicates if console logs will be saved to file	
Maximum size of console log file in MB [1-5]	3
Maximum size of console log file in MB [1-5]	
Create new log file if max size exceeded	Overwrite
Indicates if new console log file should be created or overwritten if old file maximum size was exceeded	
Path to the log directory	.\log
Directory where log files will be stored	
Check updates on startup	On
Sets automatic update check during software startup	
Check unsaved apps in devices on closing	On
Shows dialog with possibility to save unsaved applications in devices when closing iSMA Tool	

Figure 53. Available Console and logs settings

The settings applicable to logs and the Console window are the following:

- Console log level: a filter defining a kind of logs to be displayed in the Console window (the listed options provide the same result as given by the letter filter, D, A, I, and E, available at the bottom of the Console window);

- Developer mode: allows hiding/showing of the D filter: Debug button in the Console window;
- Save console logs: turns on/off recording of the log to a file saved on a hard drive;
- Maximum size of console logs file: defines the maximum size of the log file;
- Create new log file if max size exceeded: a switch defining how the system acts after exceeding the maximum log file size;
- Path to the log directory: allows defining of the log file saving location.

5.7.1 iSMA Tool Log Viewer

The iSMA Tool allows to view a file or files with logs saved on the hard drive. To open this view go to the main menu View-> Log Viewer (see the figure below).

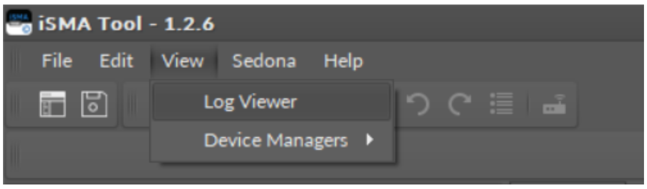
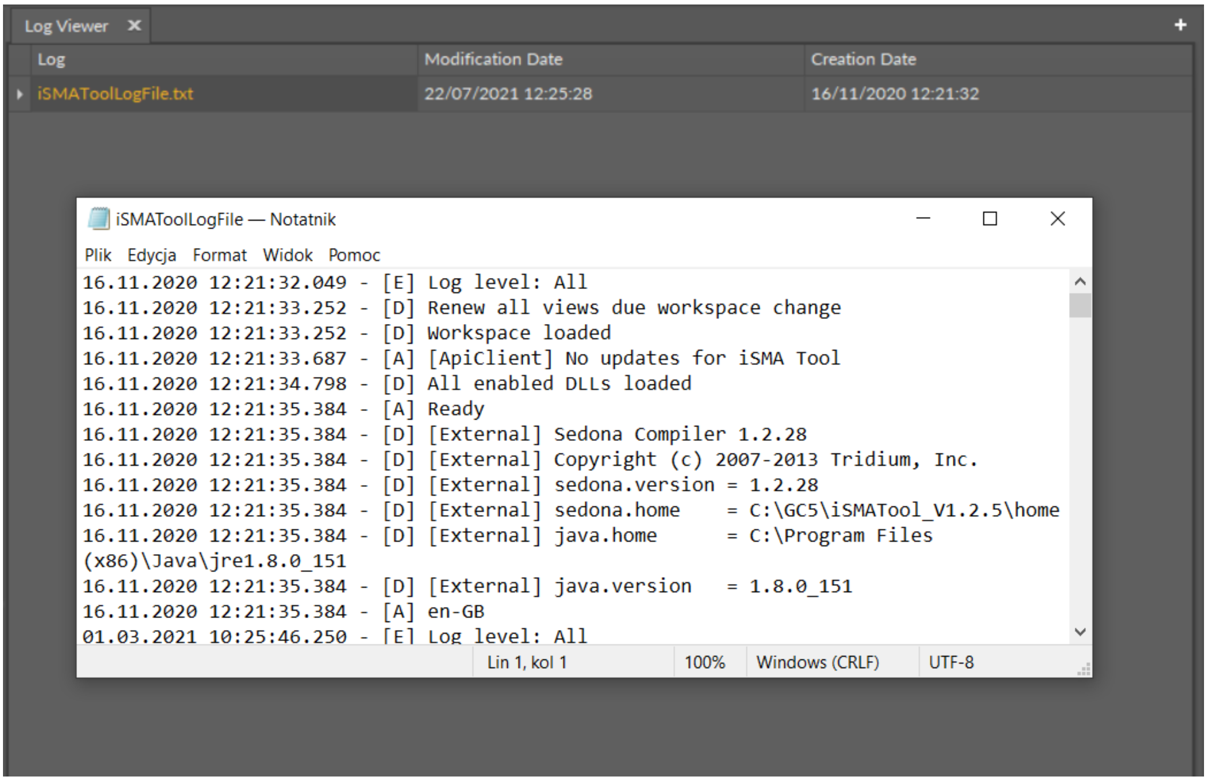


Figure 54. Accessing the Log Viewer

The Log Viewer opens in the Main screen. Once the listed log file is double-clicked, it is opened in the default .txt program.



Log	Modification Date	Creation Date
iSMAToolLogFile.txt	22/07/2021 12:25:28	16/11/2020 12:21:32

```
16.11.2020 12:21:32.049 - [E] Log level: All
16.11.2020 12:21:33.252 - [D] Renew all views due workspace change
16.11.2020 12:21:33.252 - [D] Workspace loaded
16.11.2020 12:21:33.687 - [A] [ApiClient] No updates for iSMA Tool
16.11.2020 12:21:34.798 - [D] All enabled DLLs loaded
16.11.2020 12:21:35.384 - [A] Ready
16.11.2020 12:21:35.384 - [D] [External] Sedona Compiler 1.2.28
16.11.2020 12:21:35.384 - [D] [External] Copyright (c) 2007-2013 Tridium, Inc.
16.11.2020 12:21:35.384 - [D] [External] sedona.version = 1.2.28
16.11.2020 12:21:35.384 - [D] [External] sedona.home    = C:\GC5\iSMATool_V1.2.5\home
16.11.2020 12:21:35.384 - [D] [External] java.home      = C:\Program Files
(x86)\Java\jre1.8.0_151
16.11.2020 12:21:35.384 - [D] [External] java.version   = 1.8.0_151
16.11.2020 12:21:35.384 - [A] en-GB
01.03.2021 10:25:46.250 - [E] Log level: All
```

Lin 1, kol 1 | 100% | Windows (CRLF) | UTF-8

Figure 55. Log Viewer

6 Single Device Functions

The iSMA Tool offers many functions for handling of components such as copying, duplication, removing, linking, performing actions, etc. These and other functions discussed in this chapter allow building an application in real time on a connected controller.

The functionalities described in this chapter are global, they apply in the following views: the Property Sheet, Wire Sheet, and Workspace Tree window. Individual operation of global functions, separately for each standard view, is described in Standard Views.

6.1 Context Menu

The context menu is displayed after selecting a component or link and right-clicking it.

The context menu content depends on the type of object that has been selected. The menu's content will be different for one component or link, different after selecting many components or many links, and different when components and links are selected together at the same time.

If using one of the options in the menu is impossible for what the user has selected, this option will be inactive.

The options available in the context menu are the following:

- Views: allows to display component's data in one of the defined standard views (Wire Sheet, Property Sheet, Slot Sheet) or in other views if they are available for a particular component;
- Actions: shows a list of actions that may be evoked for the given component such as Override or Set actions;
- Copy: (shortcut Ctrl + C) remembering and copying of selected component along with all its properties, settings, and links information;
- Paste: (shortcut Ctrl + V) pasting of previously remembered component into a specific place and possible recreation of internal links;
- Paste Special: same as the Paste option, plus recreating of incoming external links, if possible;
- Duplicate: (shortcut Ctrl + D) duplication of selected components in the same location;
- Delete: (shortcut Del) removal of selected components;
- Link Mark: defining the component from which a link will be led;
- Link Form: defining the component to which a link will be led;
- Reorder: changing of the order of components within a parent component;
- Rename: changing a component's name.

The figure below shows the context menu, which appears after right-clicking a single component. Options Paste and Paste Special are inactive because a Copy option has not been used. Similarly, the Link From option is inactive because the source component for a link has not been defined by a Link Mark option.

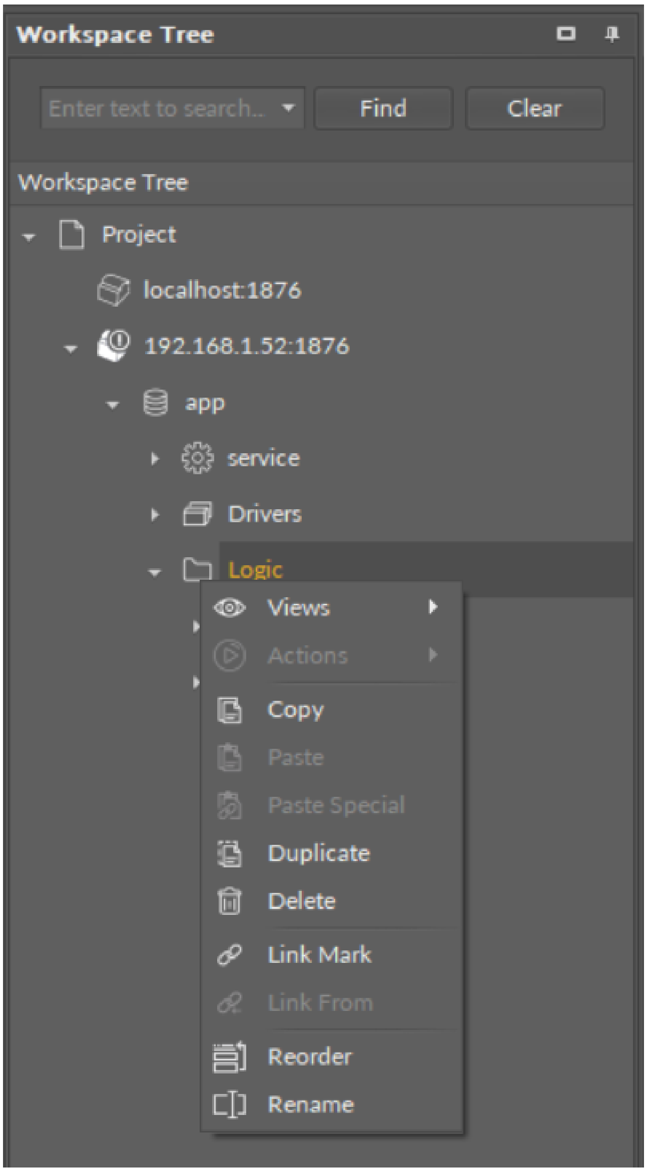


Figure 56. A context menu of a single component

The context menu visible below appears after right-clicking a link. Apart from removing it, no other operation can be performed on a link.

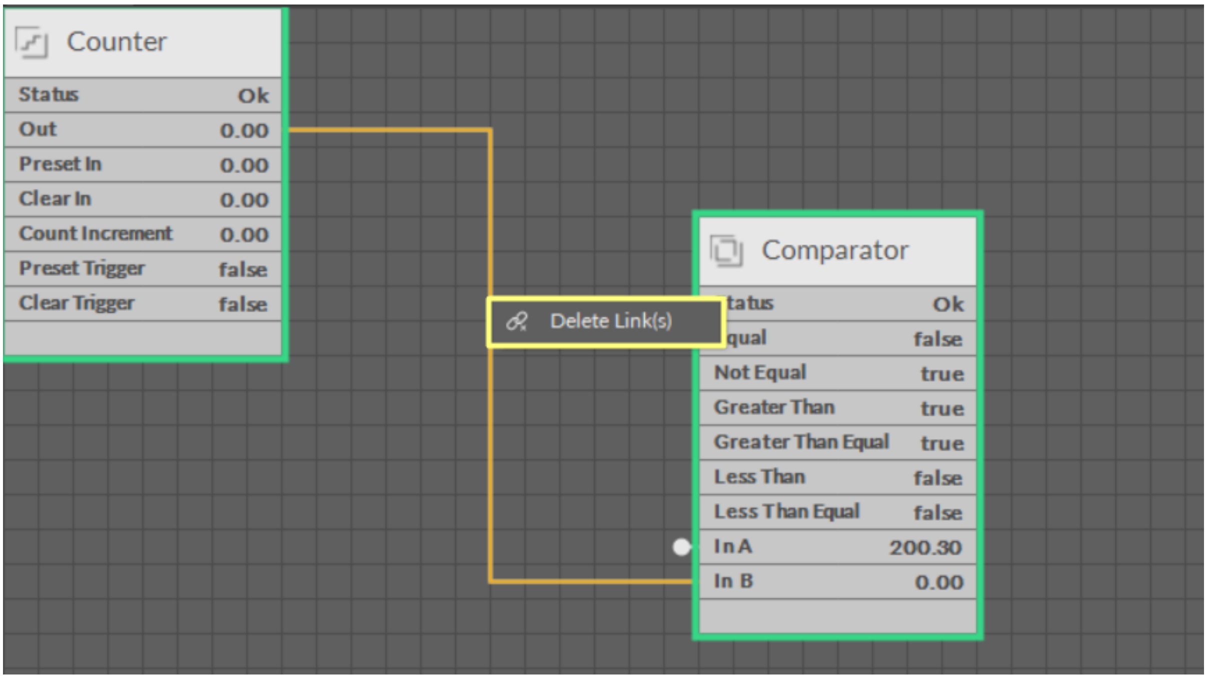


Figure 57. A context menu of a link

After selecting many components or mixed objects (component with link) most of the options are inactive because the iSMA Tool cannot perform operations on many objects simultaneously. Options available in this case will be Copy, Duplicate, and Delete, which will be performed on all selected objects.

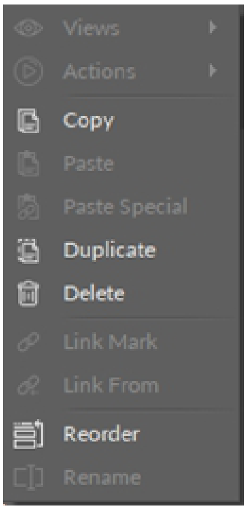


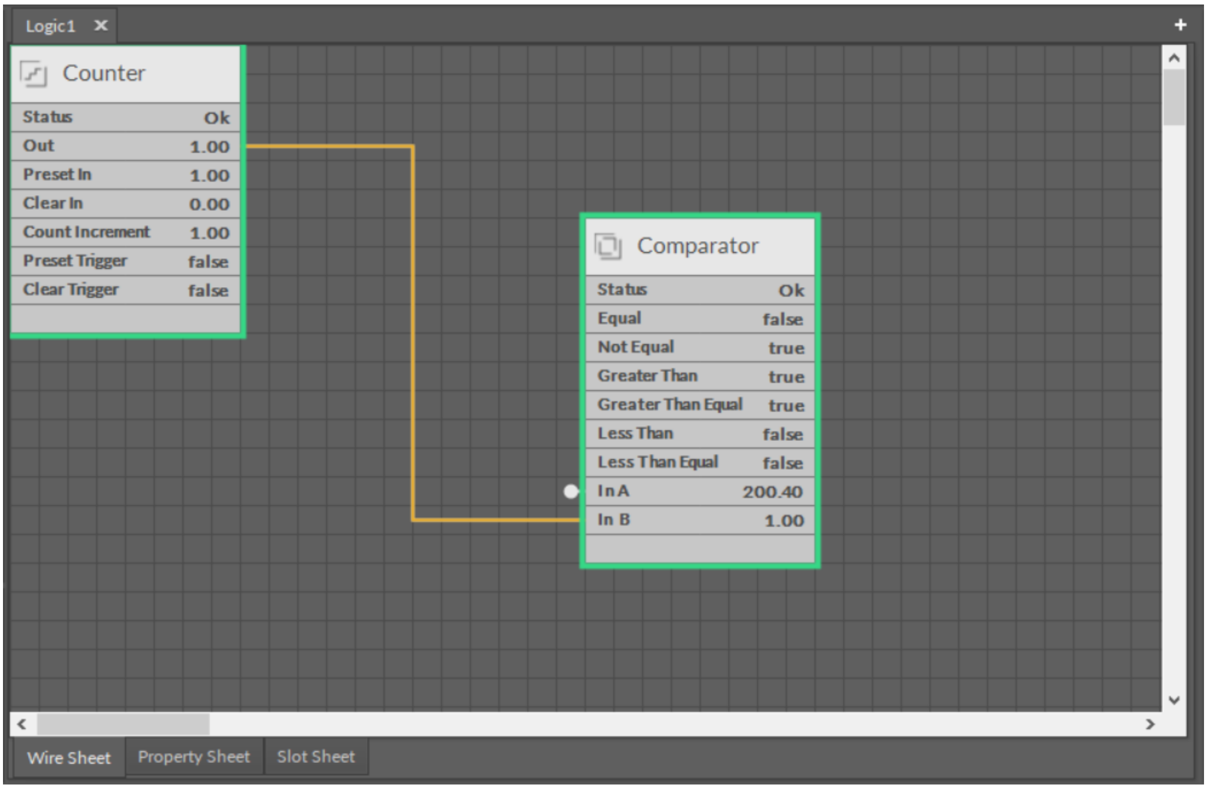
Figure 58. A context menu of multiple components

6.2 Component

A component is the smallest logical and functional unit that requires entering specific data to perform properly.

The component may be, for example, an arithmetic function of adding two numbers, therefore, it requires information about the numbers to be added. These data may be provided, for example, by other components able to read physical universal inputs of a controller, and transfer numeric data to be added.

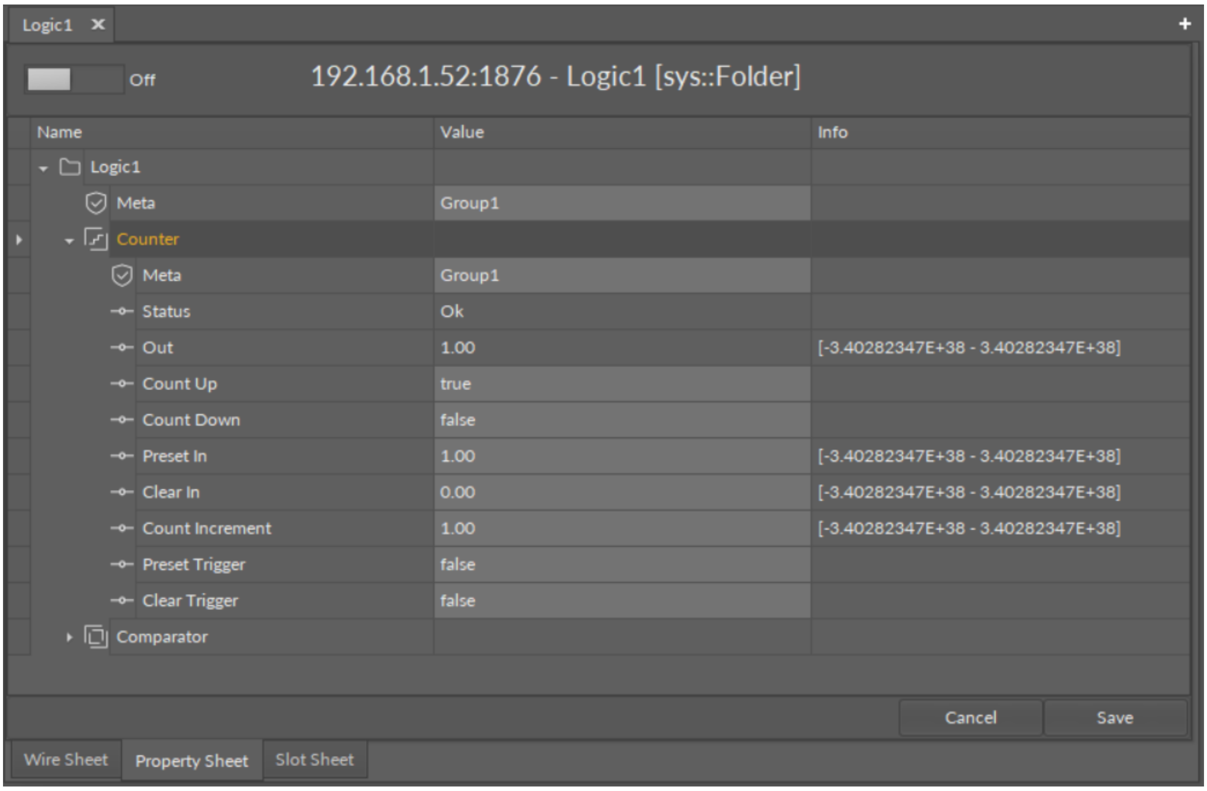
In various iSMA Tool views the component is presented differently, showing various information, as depicted on the figures below. In the Wire Sheet view the component is shown as a rectangle with a limited amount of data, and in the Property Sheet view it is a table which lists a large amount of data. Some of the slots in the table may be edited directly. More about the Property Sheet view and the Wire Sheet view may be found in the section Standard Views.



Counter
Status Ok
Out 1.00
Preset In 1.00
Clear In 0.00
Count Increment 1.00
Preset Trigger false
Clear Trigger false

Comparator
Status Ok
Equal false
Not Equal true
Greater Than true
Greater Than Equal true
Less Than false
Less Than Equal false
In A 200.40
In B 1.00

Figure 59. Components view in the Wire Sheet



192.168.1.52:1876 - Logic1 [sys::Folder]

Name	Value	Info
Logic1		
Meta	Group1	
Counter		
Meta	Group1	
Status	Ok	
Out	1.00	[-3.40282347E+38 - 3.40282347E+38]
Count Up	true	
Count Down	false	
Preset In	1.00	[-3.40282347E+38 - 3.40282347E+38]
Clear In	0.00	[-3.40282347E+38 - 3.40282347E+38]
Count Increment	1.00	[-3.40282347E+38 - 3.40282347E+38]
Preset Trigger	false	
Clear Trigger	false	
Comparator		

Figure 60. Components view in the Property Sheet

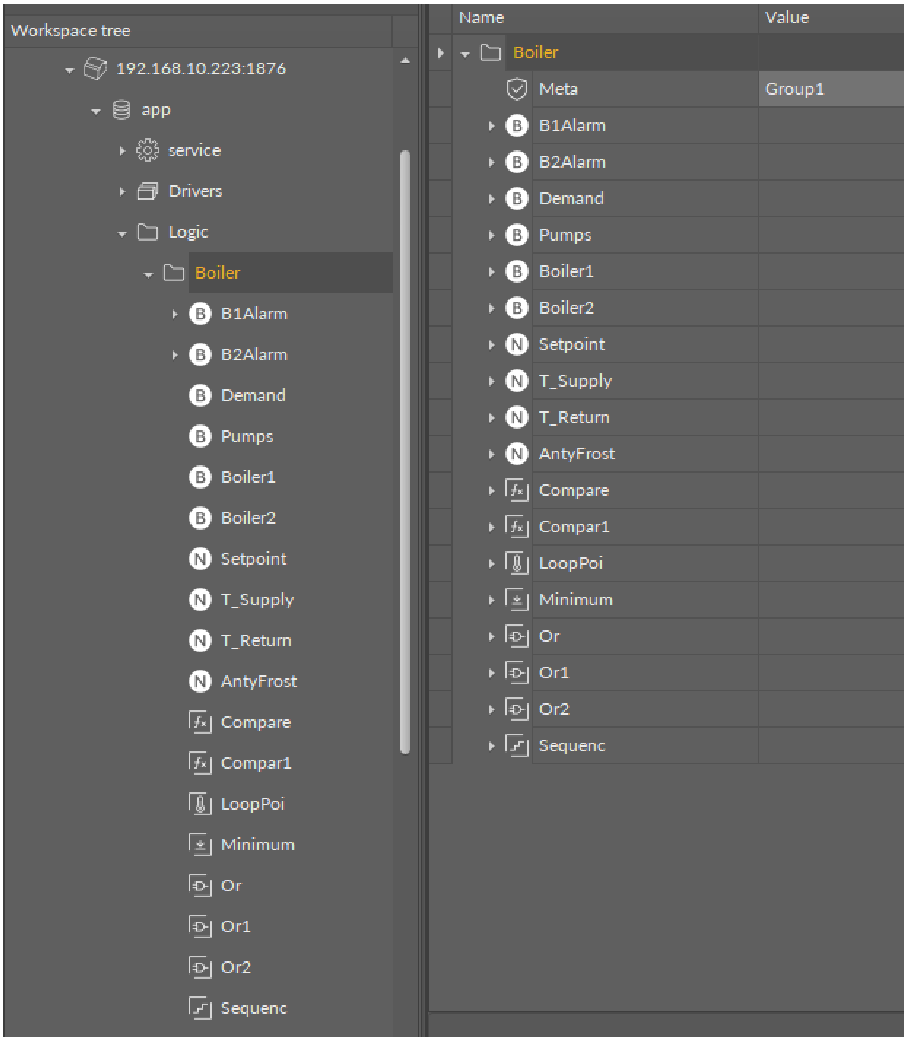
The component is the main element of application created in a controller. Therefore the iSMA Tool guarantees a standardized way of issuing commands to components. Regardless of whether the component is shown in the Wire Sheet view, Property Sheet view, or Workspace Tree window, operations are performed on it in the same way.

6.2.1 Components' Hierarchy

Components may be organized into a hierarchy tree. The component which has other components below it is called a parent, and the elements below–children. Hierarchy allows to divide a program logically, making some components dependent on the others.

In the iSMA Tool hierarchy is presented in the Workspace Tree window and on the Property Sheet view as a tree, whose superior element is a parent and inferior elements are children.

The components' hierarchy on the Wire Sheet view is limited to one generation. After opening of a parent component, the Wire Sheet view shows a schematic with all the children components distributed in the view. The figure below shows an opened parent component Boiler with its children in the Property Sheet view on the left and in the Wire Sheet view on the right.



Workspace tree

- 192.168.10.223:1876
  - app
    - service
    - Drivers
    - Logic
      - Boiler
        - B1Alarm
        - B2Alarm
        - Demand
        - Pumps
        - Boiler1
        - Boiler2
        - Setpoint
        - T_Supply
        - T_Return
        - AntyFrost
        - Compare
        - Compar1
        - LoopPoi
        - Minimum
        - Or
        - Or1
        - Or2
        - Sequenc

Name	Value
Boiler	
Meta	Group1
B1Alarm	
B2Alarm	
Demand	
Pumps	
Boiler1	
Boiler2	
Setpoint	
T_Supply	
T_Return	
AntyFrost	
Compare	
Compar1	
LoopPoi	
Minimum	
Or	
Or1	
Or2	
Sequenc	

Figure 61. Components' hierarchy on the Property Sheet view

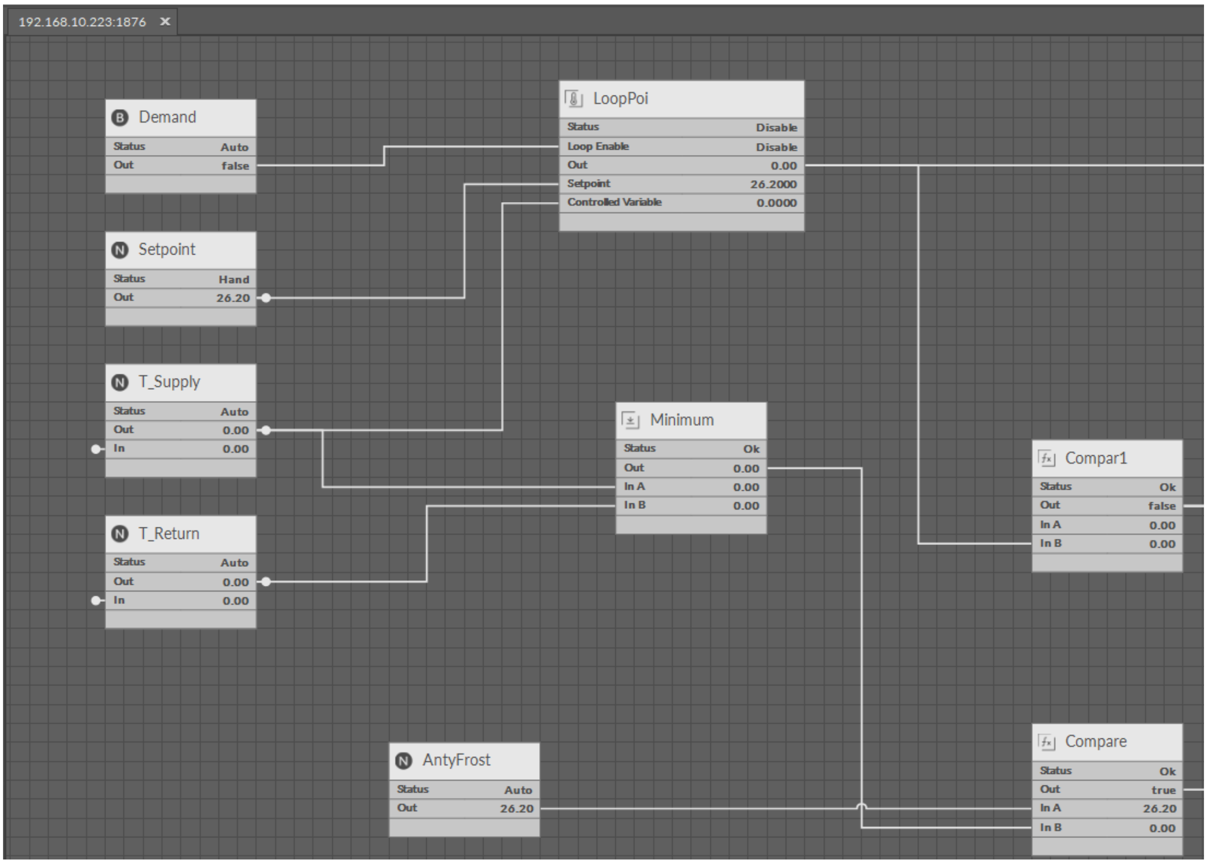


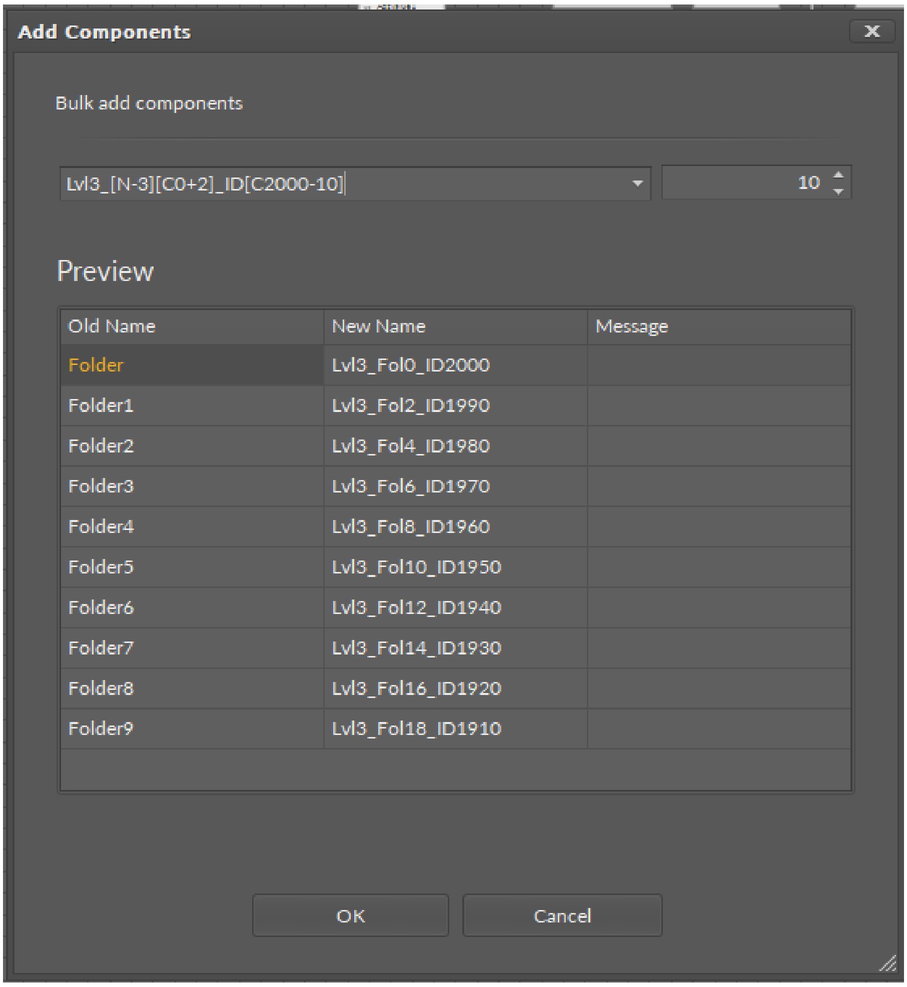
Figure 62. Components' hierarchy on the Wire Sheet view

6.2.2 Adding New Component

A new component may be created by:

- duplicating the existing component (using Copy or Duplicate function) or
- dragging a new component directly from the Device Kits window.

Each of the above methods creates a new component. The name of the new component is given automatically; however, it can be changed manually by the user. It is possible to bulk add components of the same type by indicating a number of components to add as along with a naming pattern in the dialog box (see more in Bulk Name Patterns). If several components of different types are added at the same time, the iSMA Tool will generate unique names to all the components added automatically.



Lvl3_[N-3][C0+2]_ID[C2000-10]   10

Old Name	New Name	Message
Folder	Lvl3_Fol0_ID2000	
Folder1	Lvl3_Fol2_ID1990	
Folder2	Lvl3_Fol4_ID1980	
Folder3	Lvl3_Fol6_ID1970	
Folder4	Lvl3_Fol8_ID1960	
Folder5	Lvl3_Fol10_ID1950	
Folder6	Lvl3_Fol12_ID1940	
Folder7	Lvl3_Fol14_ID1930	
Folder8	Lvl3_Fol16_ID1920	
Folder9	Lvl3_Fol18_ID1910	

Figure 63. The bulk adding dialog window

6.2.3 Copying, Pasting, and Duplication

Copying and Pasting

Copying in the iSMA Tool uses a standard Windows clipboard and may take place within the limits of one controller or between two controllers. Copying always includes all the inferior components of the selected component.

Copying Between Two Controllers

Copying between two controllers is available for AAC20 controllers. Before copying components between two controllers, make sure that a target controller (where components will be pasted to) has the same kits installed as a source controller. Otherwise, a notice window pops up, informing that copying and pasting between two controllers is not feasible.
This function is available from iSMA Tool 1.2.6 version and is independent from a firmware version of AAC20 controllers.

- Pasting copied components creates new components along with their children and slot settings effective at the moment of copying. The pasted slot value for the new component depends on a slot type: values are copied for slots of a config type, while others are reset to respective defaults.

- The iSMA Tool allows copying only of components; copying and pasting of links alone is impossible. Information about all component's links is stored in the component itself. It means that if a component is copied and pasted and there is a possibility to recreate its links from a logical point of view, the iSMA Tool will do it automatically adding new links between components.
- Copying in the iSMA Tool requires to identify components to be copied by selecting them. When a Copy option or Ctrl+C combination is used, the elements are saved to the Windows clipboard. This allows the Paste option to be used in any moment in the future until the next element is copied, even after the components, which were copied, were deleted because the iSMA Tool automatically recreates the copied structure by recreating the needed components remembered in the clipboard.
- Copying includes all the links created between the copied components. This means that while pasting the links between these components will be recreated. The situation is different for external links (links to components placed under a different parent). Basic Paste function does not support the recreation of external links, this needs another Paste Special function described in the following point.
- Copied components may be pasted into a chosen place in the Wire Sheet or Property Sheet views or in the Workspace Tree window, by indicating the pasting place and pressing Ctrl + V, or Paste from the context menu.
- Pasting function can be used many times because pasting does not erase the clipboard. The clipboard is wiped after device disconnection. The pasting option can be performed only within the same device.
- When pasting a single component, the system will ask for a new component name (with a highlighted name suggestion and support for bulk naming patterns, see Renaming Components) and number of copies to paste, and for a multi-component operation the iSMA Tool will use a mechanism ensuring that the names will be unique.

Figures below show operation of Copy and Paste functions, where newly created components (on the right figure, lower part) retained internal links.

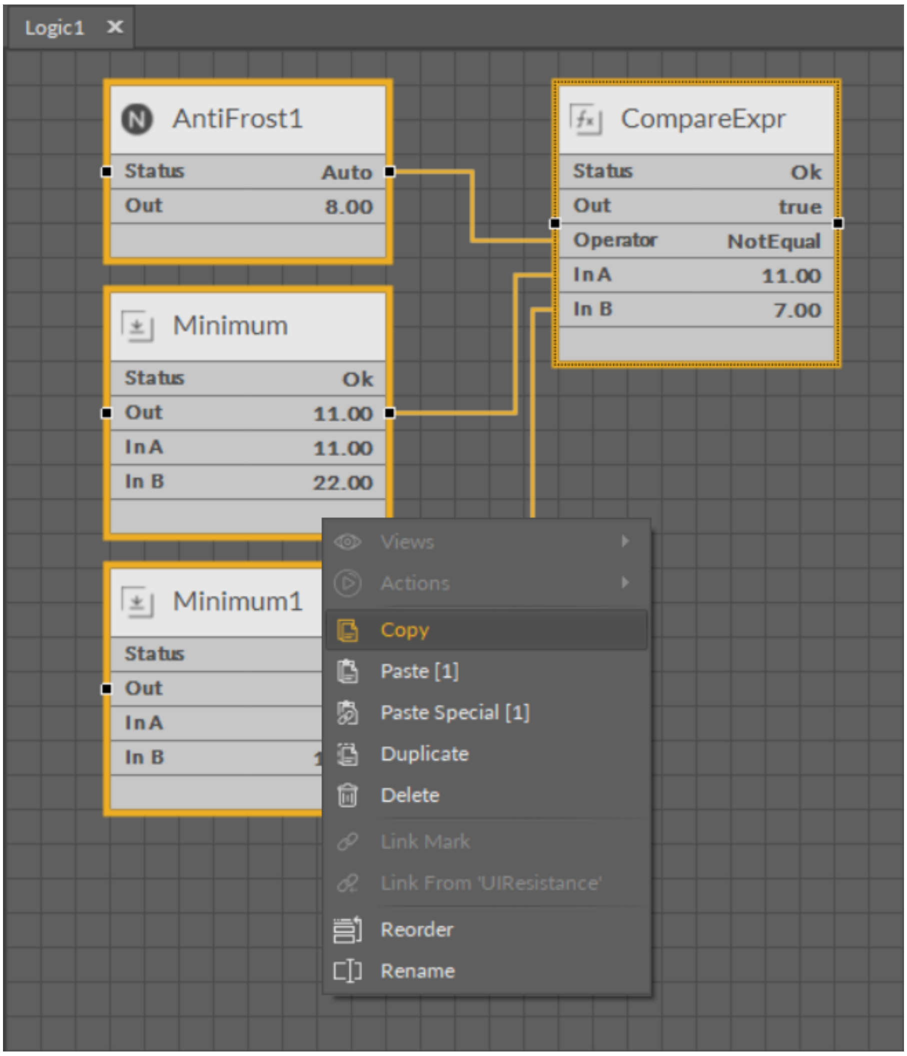
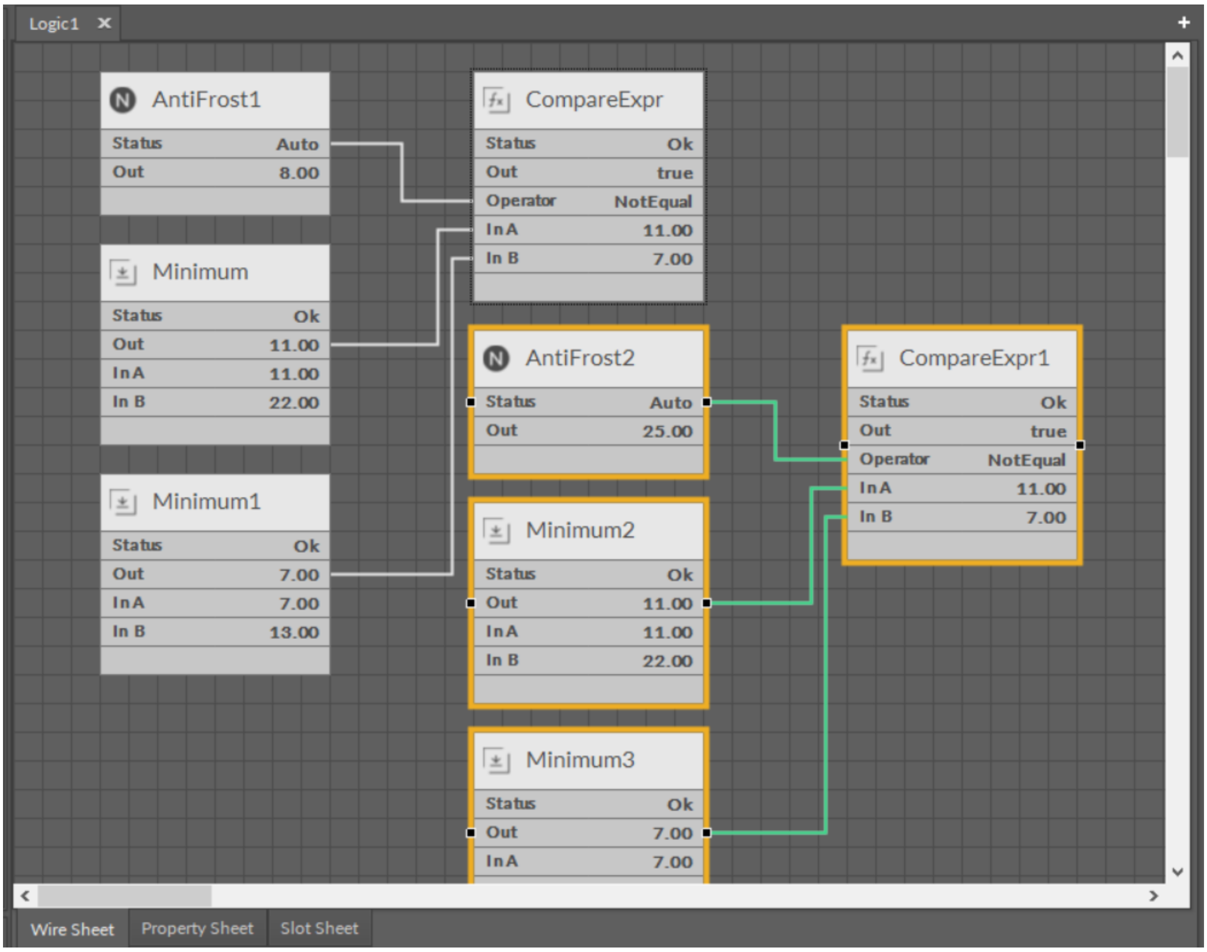


Figure 64. Coyping



AntiFrost1
Status Auto
Out 8.00

Minimum
Status Ok
Out 11.00
In A 11.00
In B 22.00

Minimum1
Status Ok
Out 7.00
In A 7.00
In B 13.00

CompareExpr
Status Ok
Out true
Operator NotEqual
In A 11.00
In B 7.00

AntiFrost2
Status Auto
Out 25.00

Minimum2
Status Ok
Out 11.00
In A 11.00
In B 22.00

Minimum3
Status Ok
Out 7.00
In A 7.00

CompareExpr1
Status Ok
Out true
Operator NotEqual
In A 11.00
In B 7.00

Figure 65. Copied and pasted components

Special Pasting

An expanded version of the Paste function is called Paste Special (available in the context menu); its aim is to recreate external links for copied components.

External links will be recreated only if all the below requirements are met:

- using the Paste Special, not the Paste, command from the context menu;
- components copied to the clipboard have external input links;
- an external component, which an external input link is recreated for, still exists.

On the figure below the CompareExpr component has been selected, and then the Copy option has been used. Only the CompareExpr component is remembered to the clipboard, even though it has both input and output external links. The figure on the right shows using a Paste Special option, where a new CompareExpr1 component retained the same settings as the original and additionally new external links were created (marked in green).

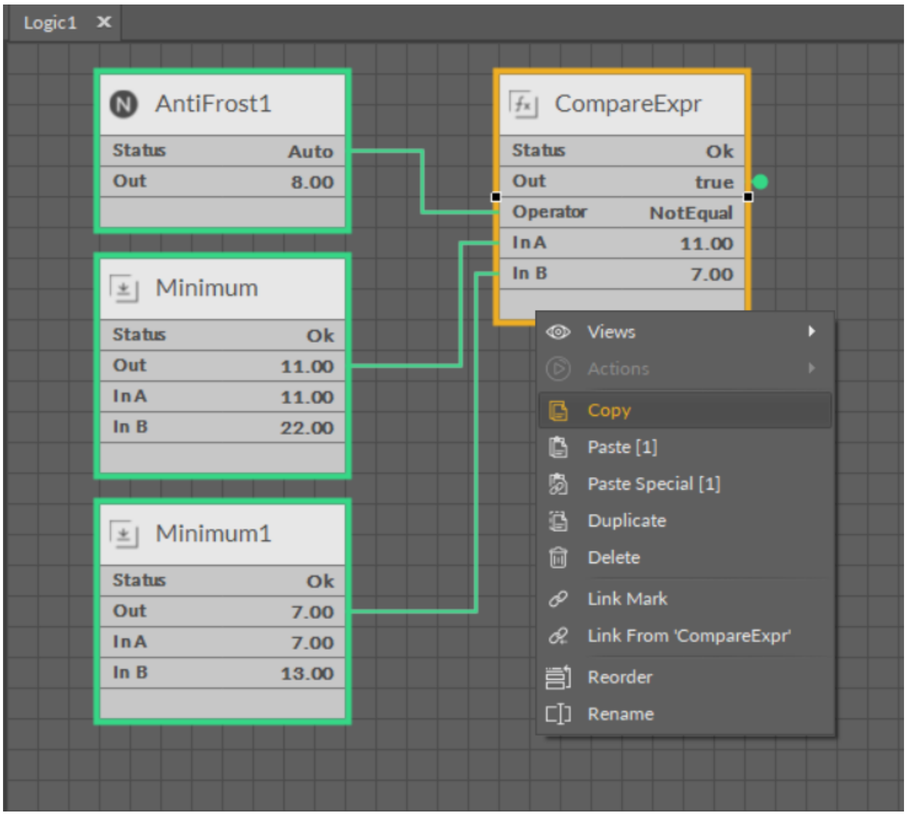


Figure 66. Copying component

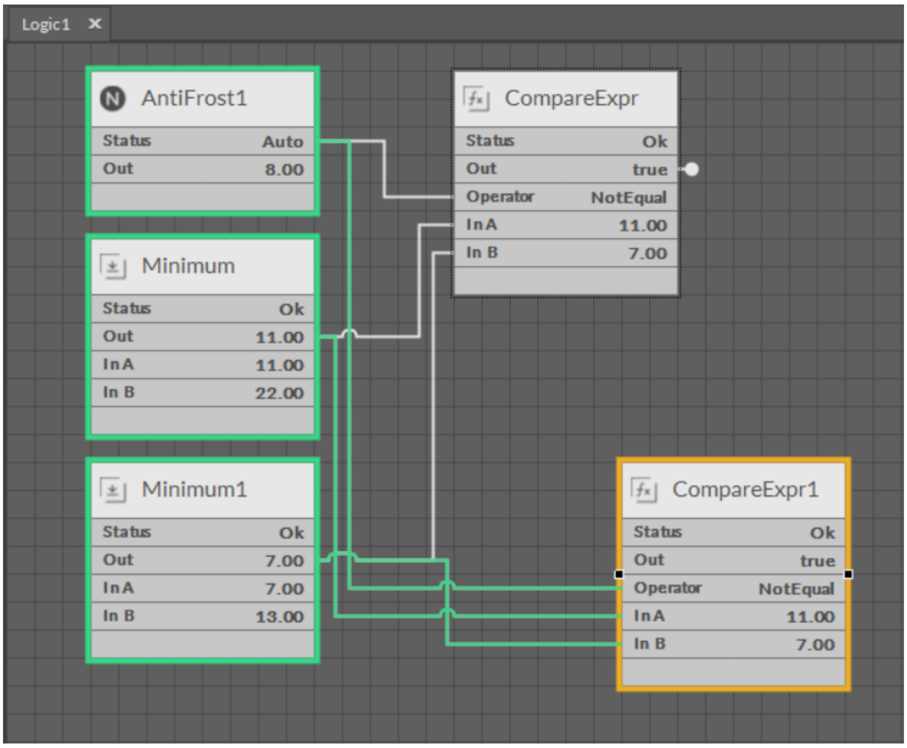


Figure 67. Special pasted component

Duplication

A Duplication is used for a quick replication of a selected component along with its values. The slot value for the duplicated component depends on a slot type: the value is copied for slots of the 'config' type, while others are reset to respective defaults.

Additionally, the duplication does not disturb what was copied by using Copy and Paste functions, i.e., it does not overwrite what is contained in the clipboard.

Duplication can be performed by key combination Ctrl+D or from the context menu function.

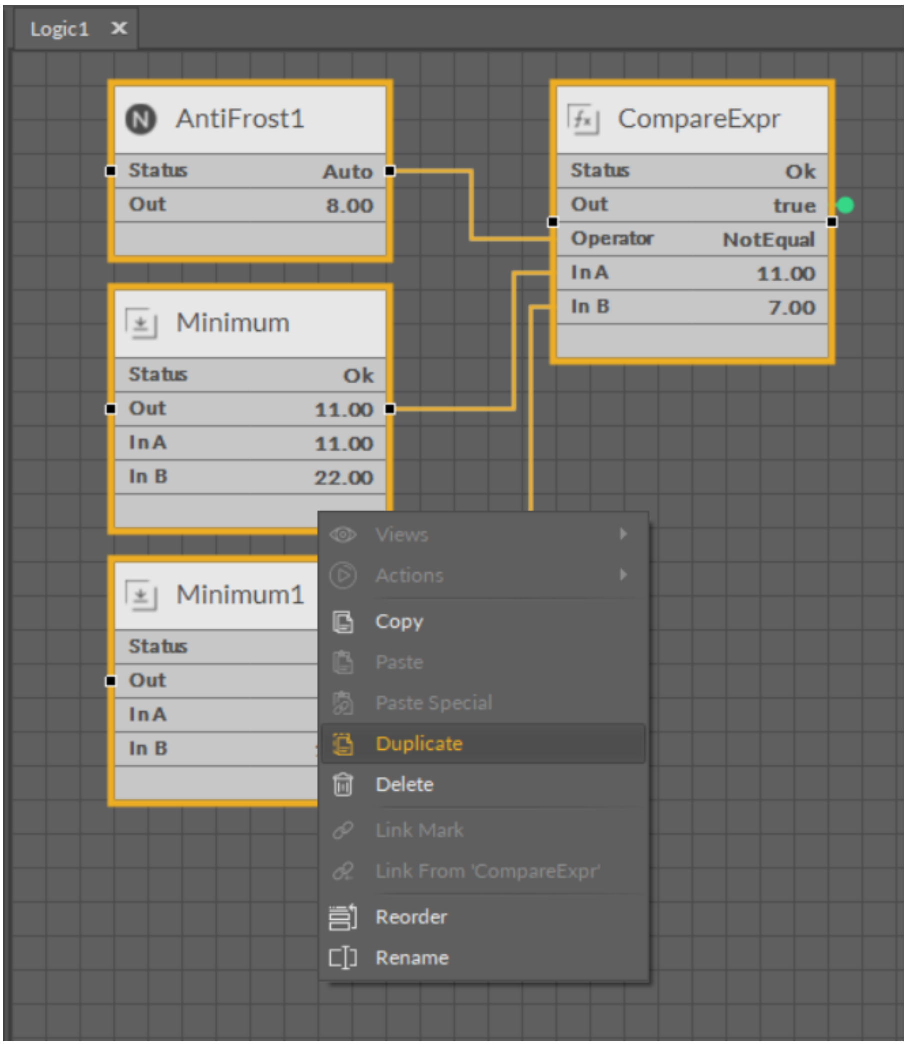
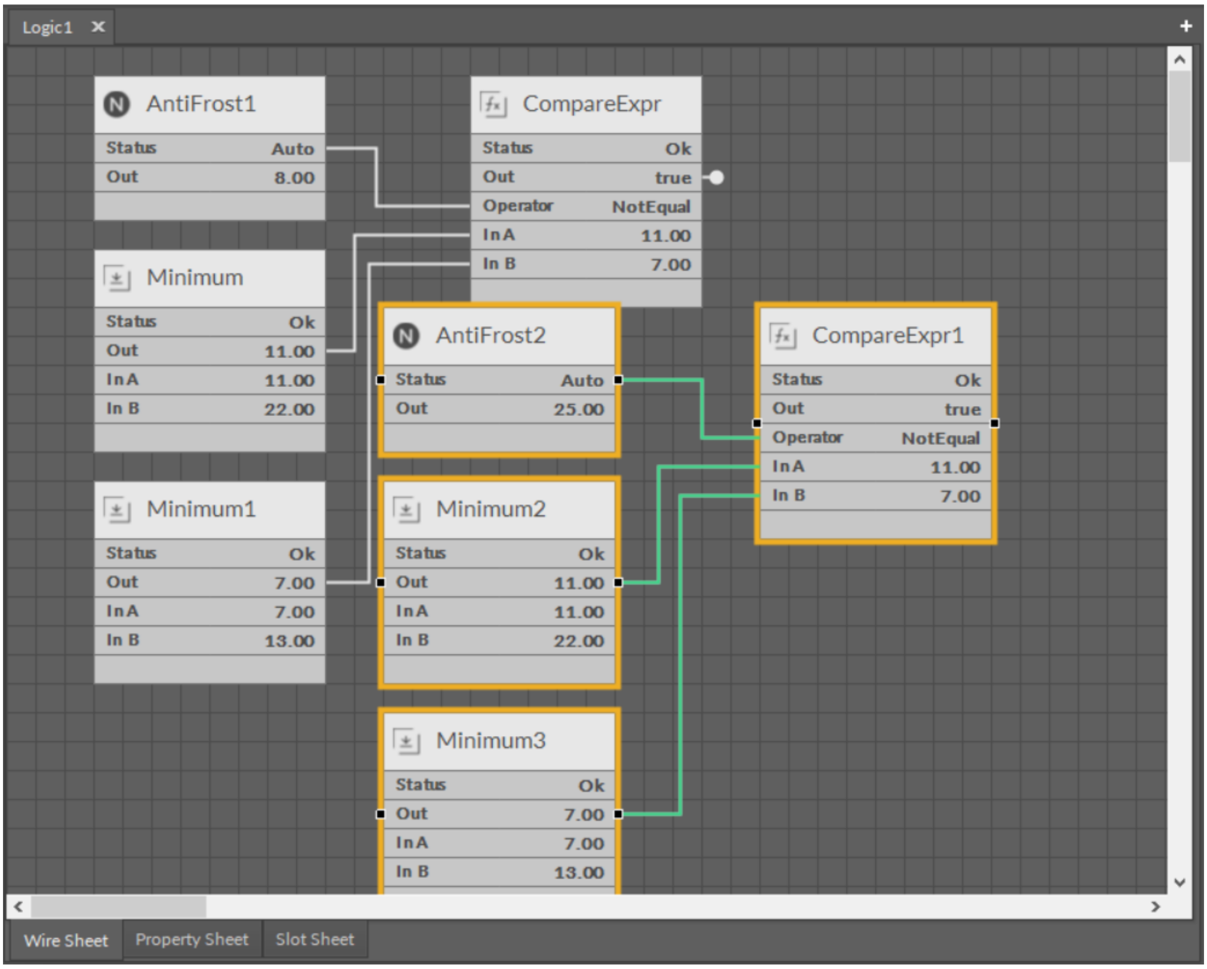


Figure 68. Duplicating components



AntiFrost1
Status Auto
Out 8.00

CompareExpr
Status Ok
Out true
Operator NotEqual
In A 11.00
In B 7.00

Minimum
Status Ok
Out 11.00
In A 11.00
In B 22.00

AntiFrost2
Status Auto
Out 25.00

CompareExpr1
Status Ok
Out true
Operator NotEqual
In A 11.00
In B 7.00

Minimum1
Status Ok
Out 7.00
In A 7.00
In B 13.00

Minimum2
Status Ok
Out 11.00
In A 11.00
In B 22.00

Minimum3
Status Ok
Out 7.00
In A 7.00
In B 13.00

Figure 69. Duplicated components

6.2.4 Linking Components

A link is a way of exchanging data between components. It connects components between which the data are exchanged. The link identifies which parameter is taken into account on both sides of the connection. This allows to show the data flow direction, i.e., where a specific parameter is read and where it is saved.

The Wire Sheet view shows the link as a line connecting two components. The topic is further discussed in Adding and Removing Links). The Workspace Tree window and the Property Sheet view do not show information about existing links at all.

Views designed to check links for a specific component are the Wire Sheet view and the Links view (available in the links tab in the Object Properties window), which will be discussed in Links Tab).

To create a link between components, in most cases a universal Link Mark method can be used, available in the context menu opened on a source component. Next, the context menu needs to be opened on a target component and the option Link From has to be chosen. That way the user will define a source and target components to create a connection (link). Now, the slots, which will send and receive data, need to be defined in both components. For this purpose, after using the Link From function, a dialog window opens allowing the user to choose a slot both on source component side as well as on target component (see the figure below).

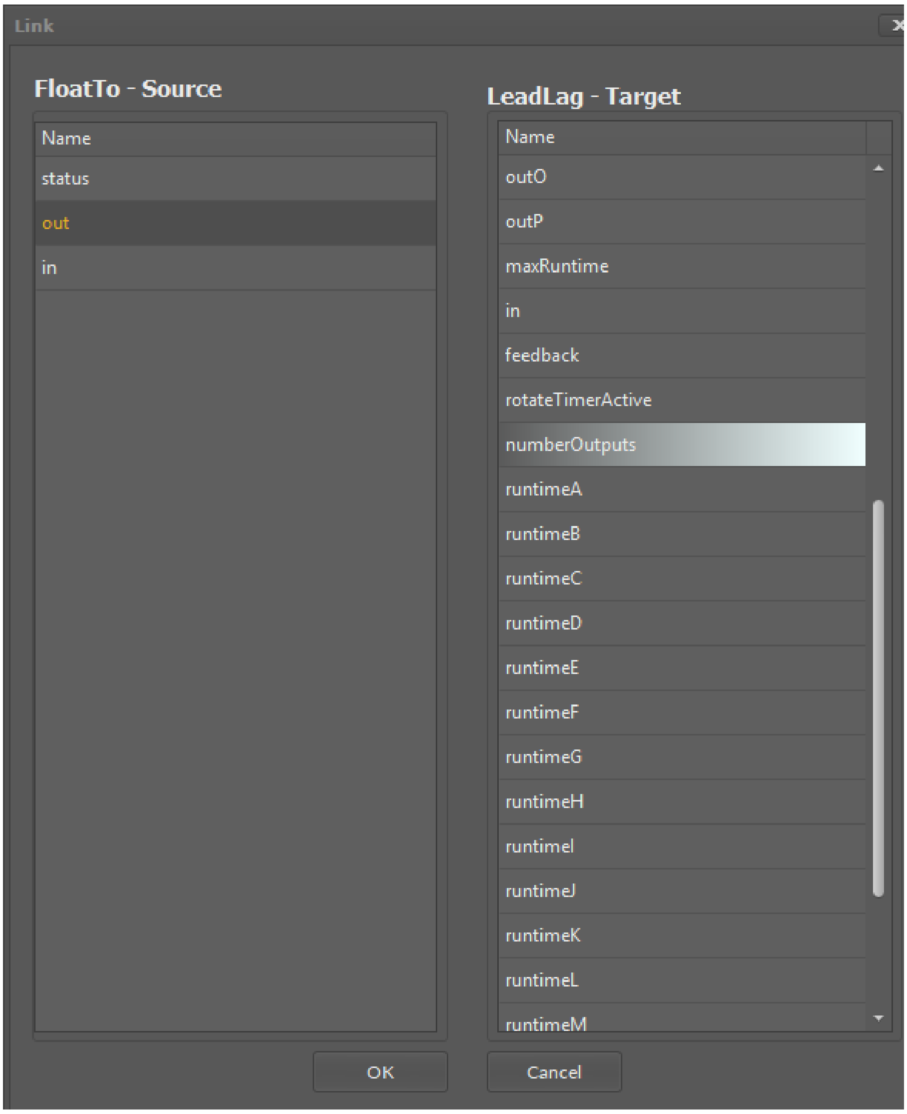


Figure 70. Dialog window for linking components

The iSMA Tool allows to link only the fitting slots of the same data type making creation of a faulty link impossible–once the source slot is selected, only the fitting target slots are highlighted.

6.2.5 Removing Components and Links

Removing Components

Deleting components allows removal of the selected components along with all their inferior components. Deleted components are instantly removed from the application in the controller, which provides full synchronization in views and windows: the Property Sheet, Wire Sheet, and Workspace Tree.

Removal is possible after selecting one or more components and pressing a Delete key on the keyboard, with the exception of removing components from the Workspace Tree.

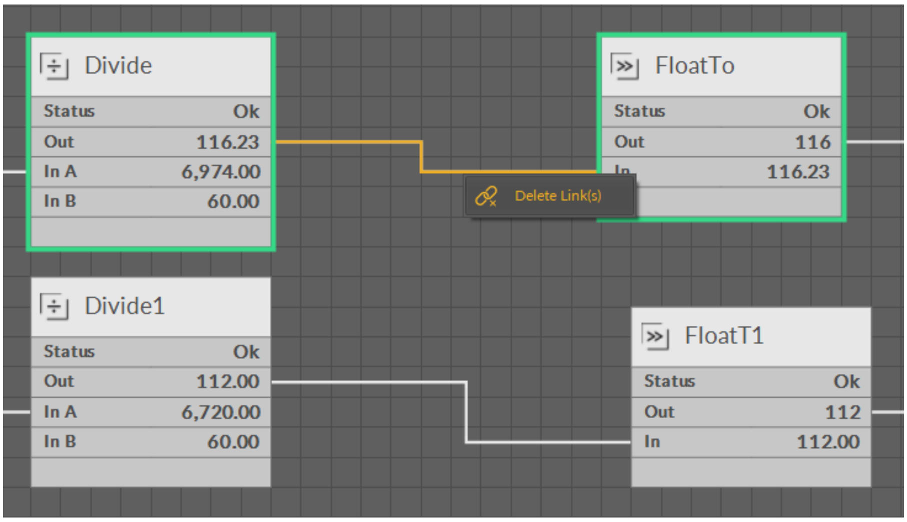
Components' removal from the Workspace Tree with the Delete key has been blocked to prevent removing the main tree elements, such as the main containers or the whole application by mistake. Removing components in the Workspace Tree is possible using the context menu but the removal requires confirming it in a pop-up window.

The above limitations do not include the Property Sheet and Wire Sheet views, where using the Delete key removes selected components instantly.

Removing Links

Removing links is performed automatically after removing the component, which a specific link leads to. It regards both input and output links.There is no need to remove links if a component is removed. However, there are situations when only the link needs to be deleted, without removing the component.

In order to remove the link only, it needs to be selected in the Wire Sheet view and then deleted by the Delete key on the keyboard or from the context menu by choosing a Delete link(s) option.



Divide
Status Ok
Out 116.23
In A 6,974.00
In B 60.00

FloatTo
Status Ok
Out 116
In 116.23

Divide1
Status Ok
Out 112.00
In A 6,720.00
In B 60.00

FloatT1
Status Ok
Out 112
In 112.00

Figure 71. Links context menu

6.2.6 Actions

Actions are the most commonly used functions possible to be performed on a specific component such as setting new value or changing the operating mode of the component into manual or automatic. These are certain operations that can be performed on components, called the actions.

Available Actions depend on the component type. It is possible to access them as following:

- Pressing the actions option on the context menu of a component;

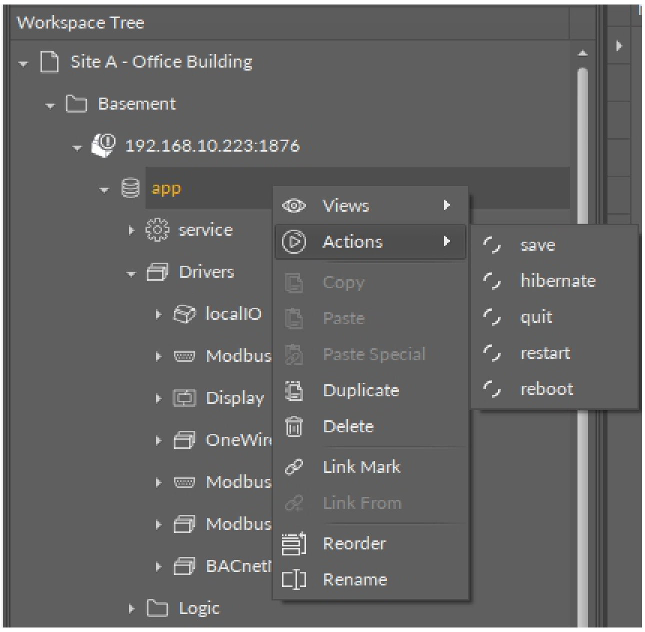
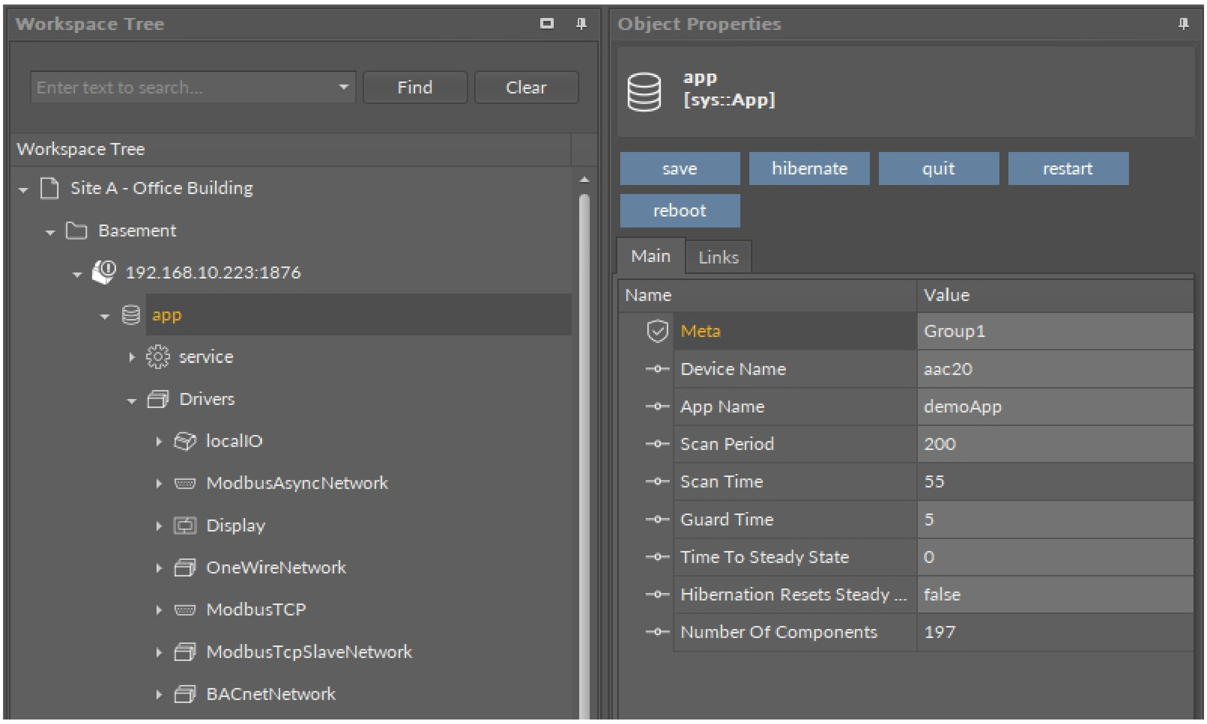


Figure 72. Accessing actions from a context menu

- Pressing the actions button available in the Object Properties window after selecting the required component.



Name	Value
Meta	Group1
Device Name	aac20
App Name	demoApp
Scan Period	200
Scan Time	55
Guard Time	5
Time To Steady State	0
Hibernation Resets Steady ...	false
Number Of Components	197

Figure 73. Accessing actions in the Object Properties window

If the component does not offer any actions, then the actions option in the context menu is disabled, as shown on the figure below.

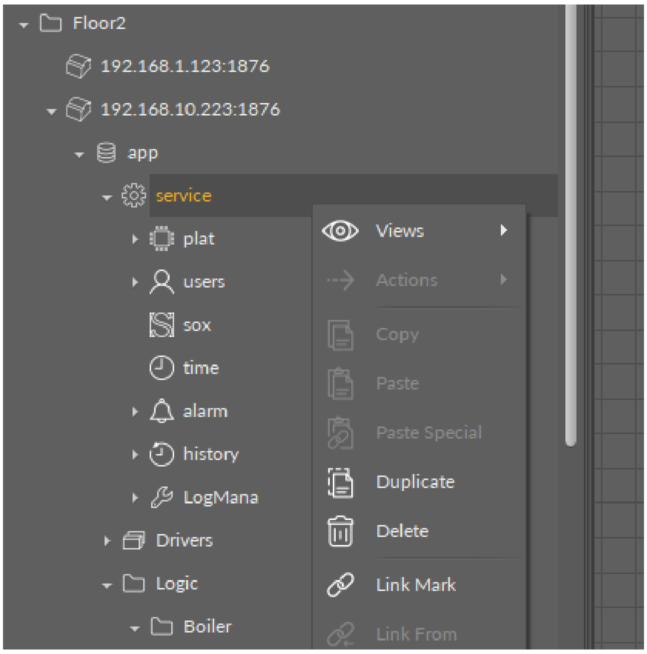


Figure 74. No available actions

If the action requires entering parameters in the component, for example, setting values for the NVNumericWriteable component type, the iSMA Tool will ask for values on a separate pop-up window (see the figure below).

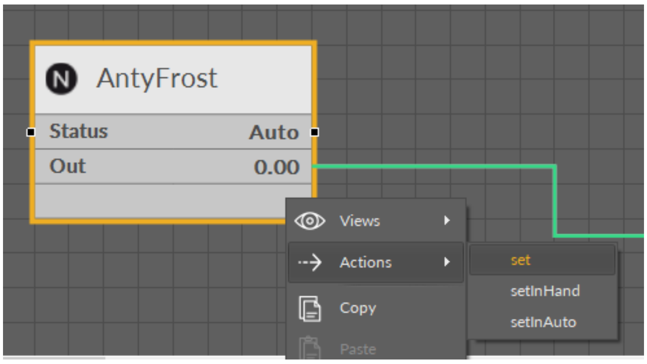


Figure 75. Setting values by actions

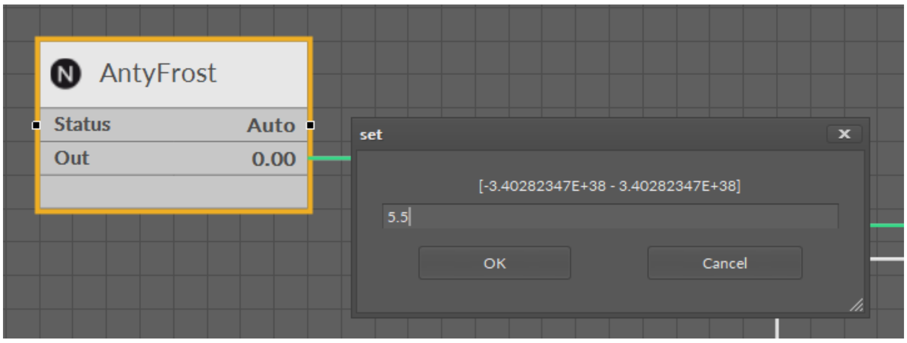


Figure 76. Setting values by actions

6.2.7 Reordering Components

While working in the iSMA Tool, the order of components under a parent component is automatically set based on the ID number. The user sees components in the order they were added to the application and there is no need to change the order to ensure proper work of the controller.

Building a text interface for the LCD display controller is an exception. In this case the order of components allowing to show text information on the LCD display is important. The order of components is the basis of the look (order) of information shown on LCD display.

In the iSMA Tool it is possible to define the order of components within a parent the following way:

In the Workspace Tree or Property Sheet view, or the Driver view (each view where the components are shown) select a component (parent) with children (other components), which will have their order changed. On the figure below the component (parent) is Local_DI.

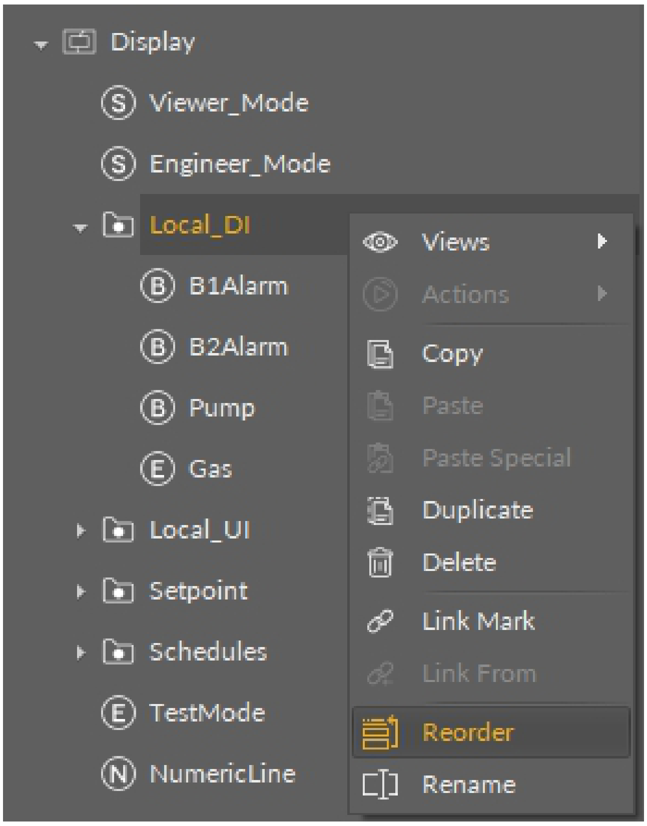
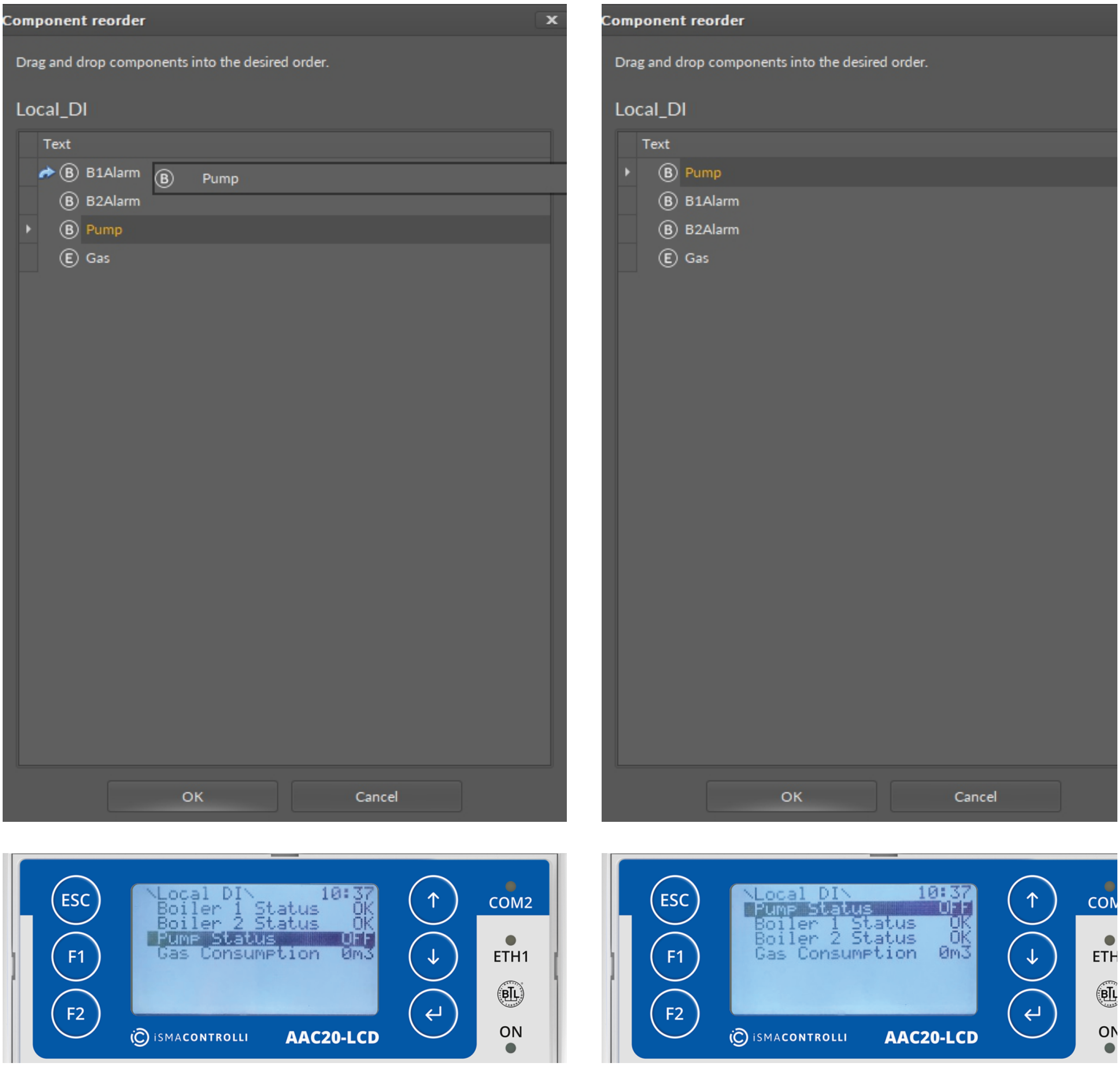


Figure 77. Reordering components from a context menu

Choose a Reorder option from the context menu, which results in opening a pop-up window allowing the order to be changed. On the figure below the Pump component is being reordered. It is moved by drag&drop to the beginning of the list.



Component reorder

Drag and drop components into the desired order.

Local_DI

Text

- B1Alarm
- B2Alarm
- Pump
- Gas

Component reorder

Drag and drop components into the desired order.

Local_DI

Text

- Pump
- B1Alarm
- B2Alarm
- Gas

```
\Local DI\        10:37
Boiler 1 Status      OK
Boiler 2 Status      OK
Pump Status         OFF
Gas Consumption     0m3
```

```
\Local DI\        10:37
Pump Status         OFF
Boiler 1 Status      OK
Boiler 2 Status      OK
Gas Consumption     0m3
```

Table 2. Reordering components effective on the AAC20 LCD display

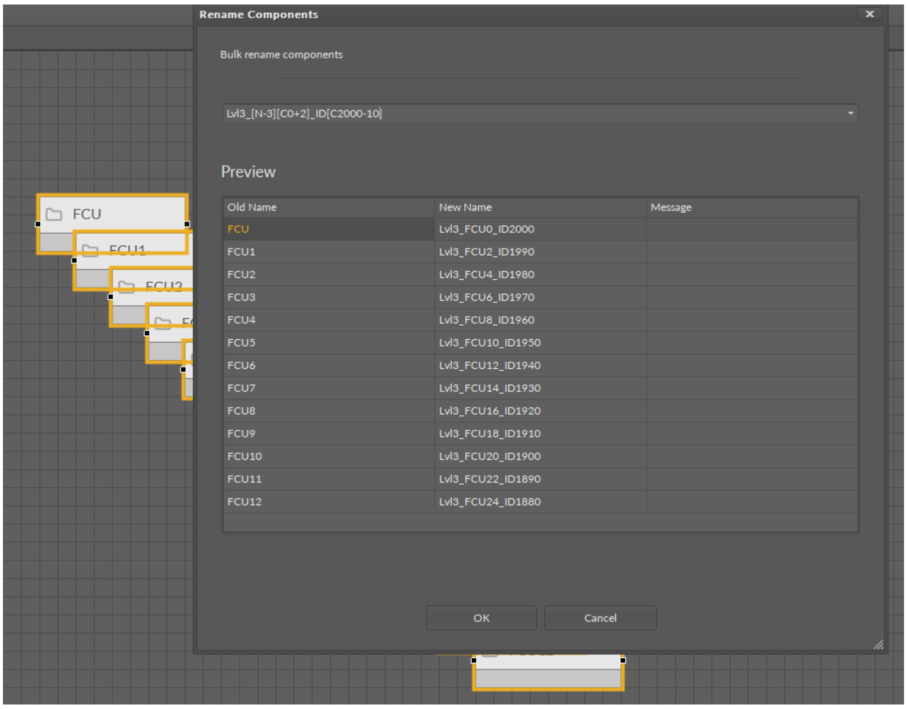
Warning!

In case of using the Reorder function on a childless component, the same Reorder function will be used for the parent of the selected component. This allows using the Reorder function by selecting one of the components which will be reordered (provided it has no children).

When the components order changes are done, they need to be confirmed in a pop-up window by clicking an OK button. New order of the components is instantly sent to the controller, which is also illustrated with the order change in the Workspace Tree and other views showing components.

6.2.8 Renaming Components

A Rename option selected on multiple components allows changing names of more than one components at once. The Rename pattern defines how the new name for components will be generated. It can include some constant literals with special placeholders embedded.



Rename Components

Bulk rename components

Lvl3_[N-3][C0+2]_ID[C2000-10]

Preview

Old Name	New Name	Message
FCU	Lvl3_FCU0_ID2000	
FCU1	Lvl3_FCU2_ID1990	
FCU2	Lvl3_FCU4_ID1980	
FCU3	Lvl3_FCU6_ID1970	
FCU4	Lvl3_FCU8_ID1960	
FCU5	Lvl3_FCU10_ID1950	
FCU6	Lvl3_FCU12_ID1940	
FCU7	Lvl3_FCU14_ID1930	
FCU8	Lvl3_FCU16_ID1920	
FCU9	Lvl3_FCU18_ID1910	
FCU10	Lvl3_FCU20_ID1900	
FCU11	Lvl3_FCU22_ID1890	
FCU12	Lvl3_FCU24_ID1880	

Figure 78. Renaming components

Given the original names of components were "FCU", "FCU1", ... , "FCU12", a possible Rename pattern may be composed as follows:

Lvl3_[N-3][C0+2]_ID[C2000-10]

to generate the following names:

FCU → Lvl3_FCU0_ID2000

FCU1 → Lvl3_FCU2_ID1990

FCU2 → Lvl3_FCU4_ID1980

...

FCU8 → Lvl3_FCU16_ID1920

FCU9 → Lvl3_FCU18_ID1910

FCU10 → Lvl3_FCU20_ID1900

FCU11 → Lvl3_FCU22_ID1890

FCU12 → Lvl3_FCU24_ID1880

The Placeholder syntax:

[N] Substring of the original name

[Nf-t] Inserts a substring of the original name starting at symbol index 'f' up to the symbol index 't'. The leftmost symbol has an index of ‘1’.

Variants:

- [N] A whole name;
- [N5] A single symbol at index 5;
- [N3-] A substring from index 3 to the end of the original name;
- [N3-5] Symbols from index 3 to 5;
- [N-6] Symbols from the beginning to index 6.
- [C] Counter
- [Cf+/-s] Inserts a counter starting at value 'f' using the step +/-s.
- Variants:
- [C] A counter starting at 1, step +1 (1,2,3,4,5,...);
- [C5] A counter starting at 5, step +1 (5,6,7,8,9,...);
- [C3+5] A counter starting at 3, step +5 (3,8,13,18,...);
- [C100-5] A counter starting at 100, step5 (100, 95, 90, 85,...).

Note: No spaces may be included in the component’s name pattern.

7 Kit Manager

The Kit Manager is a specific tool in the device allowing the management of its Kits. The Kit Manager allows installing and uninstalling of kits as well as changing versions of kits already existing in the device.

The Kit Manager allows indication of actions which are to be performed on kits (adding, removing, version change) and performing all actions at once after giving a command to update the device's kits.

Warning!

Every time kits are edited in the device, it requires a restart of the device.

Warning!

Working with the Kit Manager for the iSMA-B-FCU controller is blocked due to the construction and purpose of the controller.

7.1 Kits Installed in iSMA Tool

The Kit Manager may be opened in two ways–defining or without defining the device the Kit Manager will cooperate with–already at the start-up stage.

Opening the Kit Manager without defining the device it will work with, allows to check what kits are currently available (installed) in the iSMA Tool, which will be available to be added to the device in the next stage.

To open the Kit Manager with a list of kits available the in iSMA Tool it is necessary to go to the main menu -> View -> Device Managers -> Kit Manager.

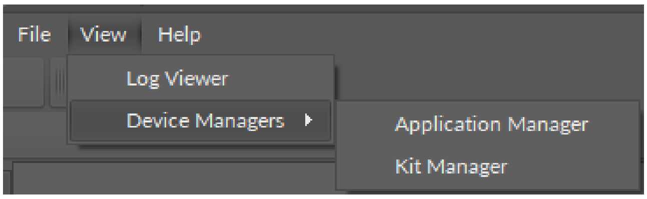
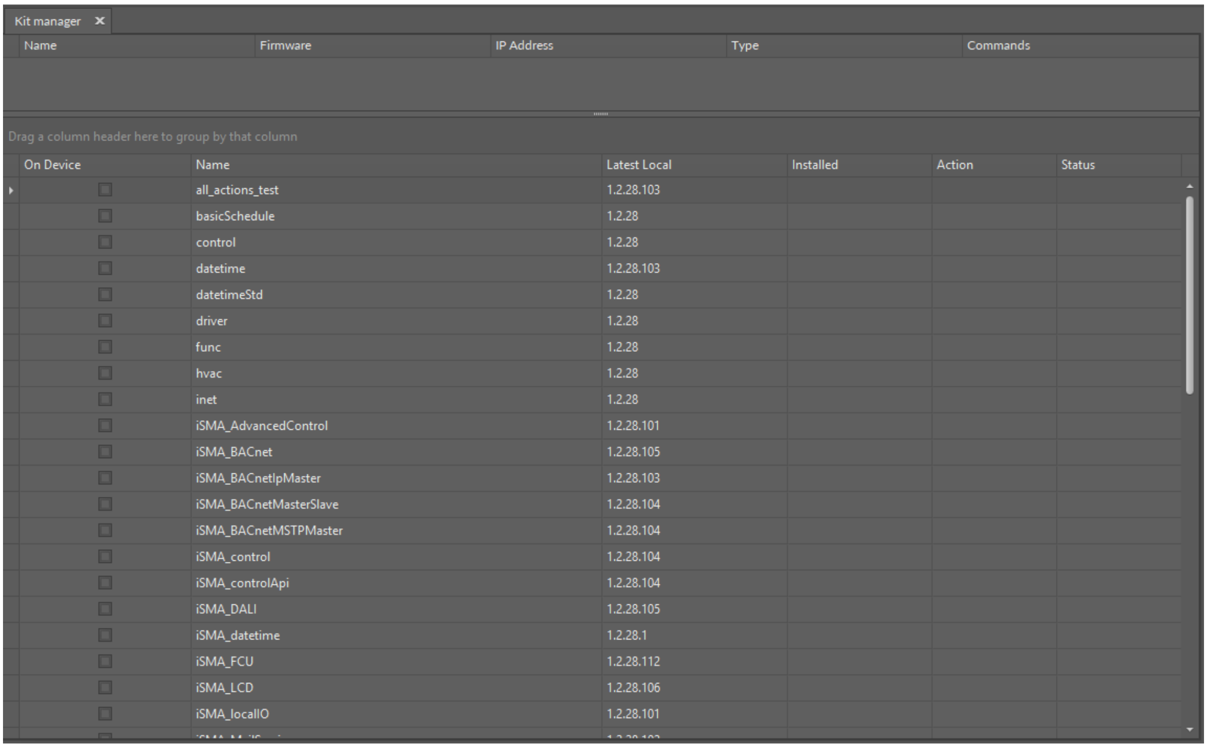


Figure 79. Accessing the Kit Manager

The appearing view consists of two sections: empty upper one for cooperating with devices and lower one with a kit list, see the figure below.



Name	Firmware	IP Address	Type	Commands

Drag a column header here to group by that column

On Device	Name	Latest Local	Installed	Action	Status
□	all_actions_test	1.2.28.103			
□	basicSchedule	1.2.28			
□	control	1.2.28			
□	datetime	1.2.28.103			
□	datetimeStd	1.2.28			
□	driver	1.2.28			
□	func	1.2.28			
□	hvac	1.2.28			
□	inet	1.2.28			
□	iSMA_AdvancedControl	1.2.28.101			
□	iSMA_BACnet	1.2.28.105			
□	iSMA_BACnetIpMaster	1.2.28.103			
□	iSMA_BACnetMasterSlave	1.2.28.104			
□	iSMA_BACnetMSTPMaster	1.2.28.104			
□	iSMA_control	1.2.28.104			
□	iSMA_controlApi	1.2.28.104			
□	iSMA_DALI	1.2.28.105			
□	iSMA_datetime	1.2.28.1			
□	iSMA_FCU	1.2.28.112			
□	iSMA_LCD	1.2.28.106			
□	iSMA_localIO	1.2.28.101			

Figure 80. Kit Manager

The lower section shows an uneditable table where every row represents a Kit installed in the iSMA Tool. The meaning of columns is as follows:

- On Device: informs whether the kit is currently installed on a chosen device. In this case the table shows only kits present in the iSMA Tool, so all the checkboxes in this column are empty;
- Name: shows a name of the kit;
- Latest Local: shows a number of the newest version of the kit installed locally in the iSMA Tool.

The list of kits visible in the table corresponds with a physical content of kits folder in the main iSMA Tool location: „/home/kits. Proper placing of user's own kit in this location allows to add it to the Kit Manager.

To update kits in a device, the device to work with has to be exactly defined.

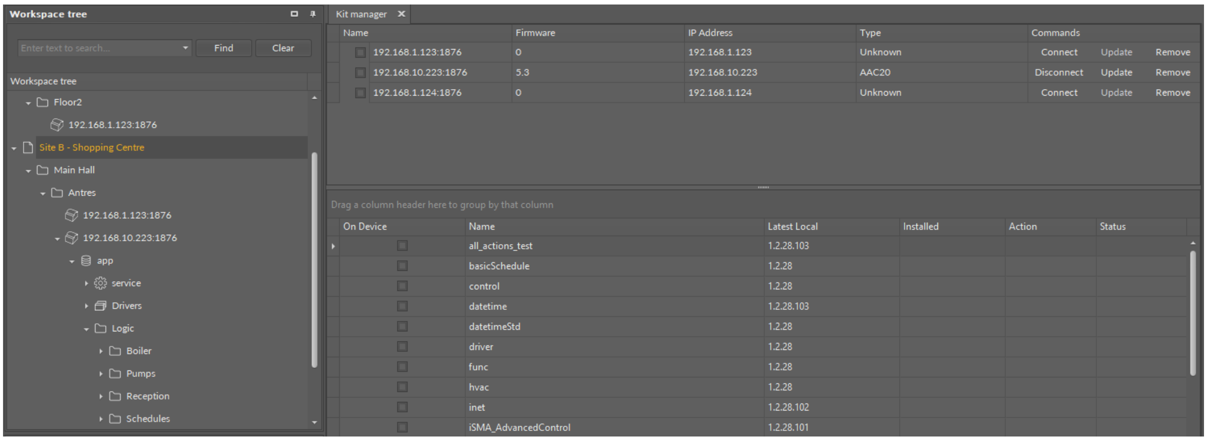
There are two methods to initiate the Kit Manager at the same time defining the device it will cooperate with. They are:

- initiating the Kit Manager from the context menu;
- initiating the Kit Manager from the Object Properties Window.

7.2 Adding Devices to Kit Manager

The function of adding the device to the Kit Manager, described below, may be used regardless of the method the Kit Manager has been initiated.

To add a single device or many devices at once to the Kit Manager, the required device needs to be dragged from the Workspace Tree into the upper part of the Kit Manager view, where the list of chosen devices is. There is also a possibility of dragging the whole subtree from the Workspace Tree with many devices at once. In this situation the iSMA Tool will filter the dragged objects and add only the devices to a view list. The figure below shows a view after dragging (adding) all the devices placed in the Site B–Shopping Centre subtree.



Name	Firmware	IP Address	Type	Commands		
□ 192.168.1.123:1876	0	192.168.1.123	Unknown	Connect	Update	Remove
□ 192.168.10.223:1876	5.3	192.168.10.223	AAC20	Disconnect	Update	Remove
□ 192.168.1.124:1876	0	192.168.1.124	Unknown	Connect	Update	Remove

Drag a column header here to group by that column

On Device	Name	Latest Local	Installed	Action	Status
□	all_actions_test	1.2.28.103			
□	basicSchedule	1.2.28			
□	control	1.2.28			
□	datetime	1.2.28.103			
□	datetimeStd	1.2.28			
□	driver	1.2.28			
□	func	1.2.28			
□	hvac	1.2.28			
□	inet	1.2.28.102			
□	iSMA_AdvancedControl	1.2.28.101			

Figure 81. Devices added to the Kit Manager

After adding the devices to the Kit Manager, it is necessary to select one of the devices, which will be operated, by ticking a checkbox. Only when the device is defined, it will be possible to use the Kit Manager for this device.

Note: The Kit Manager allows to work with one device at a time. The device list in the upper part of the Kit Manager is used for an easy switching between devices.

7.3 Initiating Kit Manager from Context Menu

Another method to add a single device to work with the Kit Manager is using the context menu, initiated on the device in the Workspace Tree window, see the figure below.

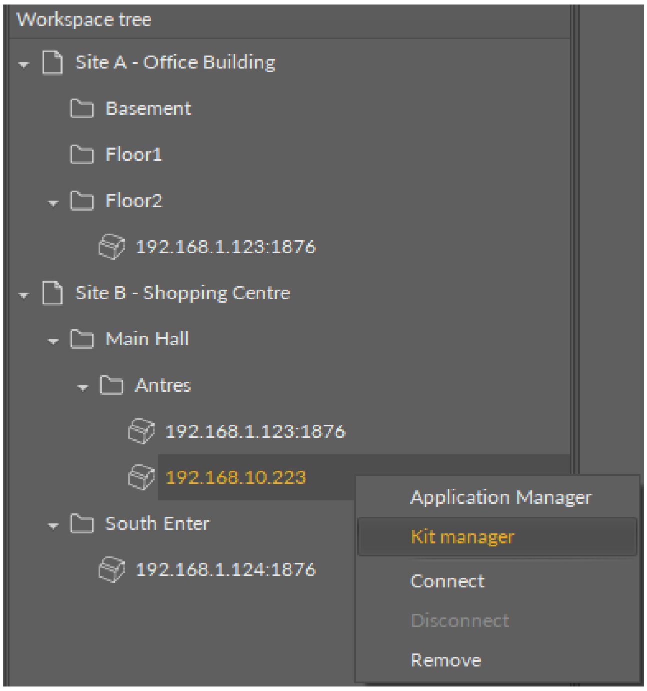
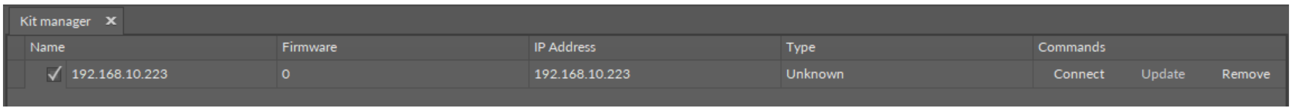


Figure 82. Initiating the Kit Manager from a context menu

After automatic opening of the Kit Manager, the device is added in the upper area of the view, as shown on the figure below.

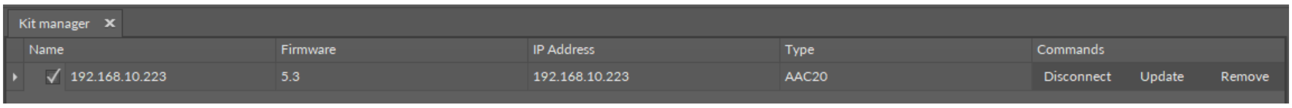


Kit manager ✕

Name	Firmware	IP Address	Type	Commands
✓ 192.168.10.223	0	192.168.10.223	Unknown	Connect Update Remove

Figure 83. Kit Manager opened for a device

If the device parameters shown in the table (such as on the figure above) are not specified, before working further with the device it needs to be connected using a Connect button. After a correct authorization and connection with the device, all the device parameters will be shown as on the figure below.



Kit manager ✕

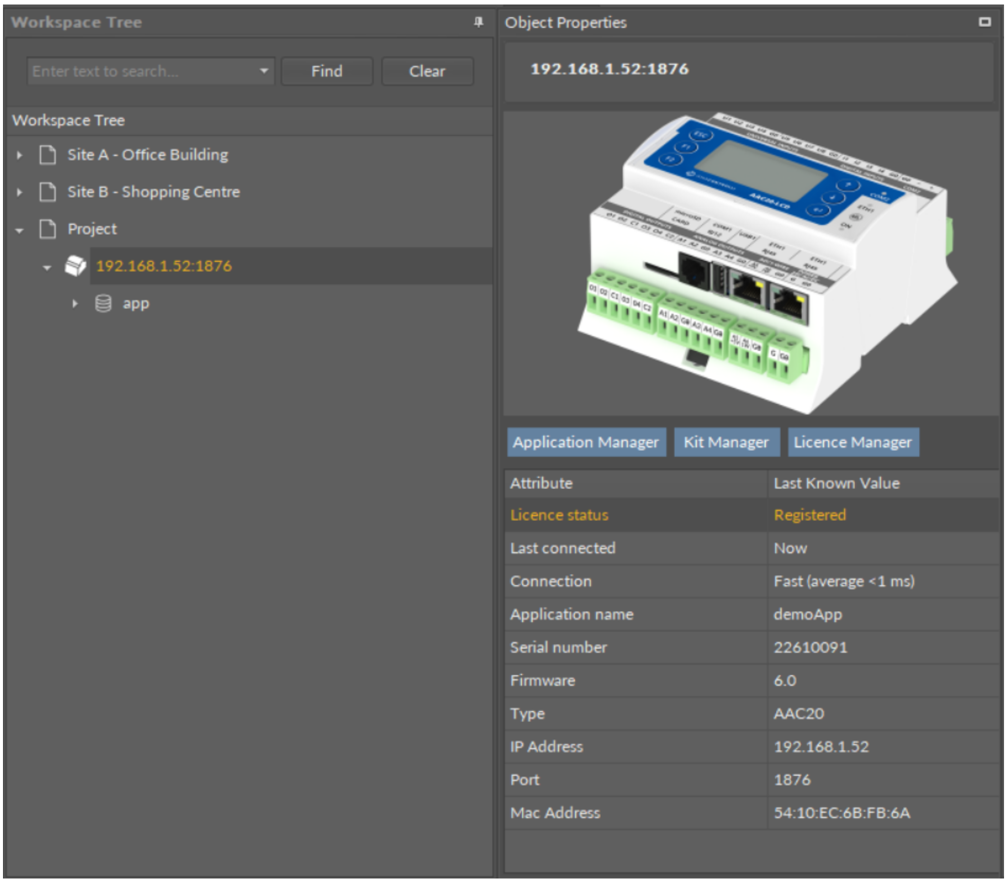
Name	Firmware	IP Address	Type	Commands
✓ 192.168.10.223	5.3	192.168.10.223	AAC20	Disconnect Update Remove

Figure 84. Kit Manager for a connected device

It is now possible for the Kit Manager to work with this device.

7.4 Initiating Kit Manager from Object Properties

Another convenient way of initiating the Kit Manager for a preselected device is using the button available in the Object Properties window.



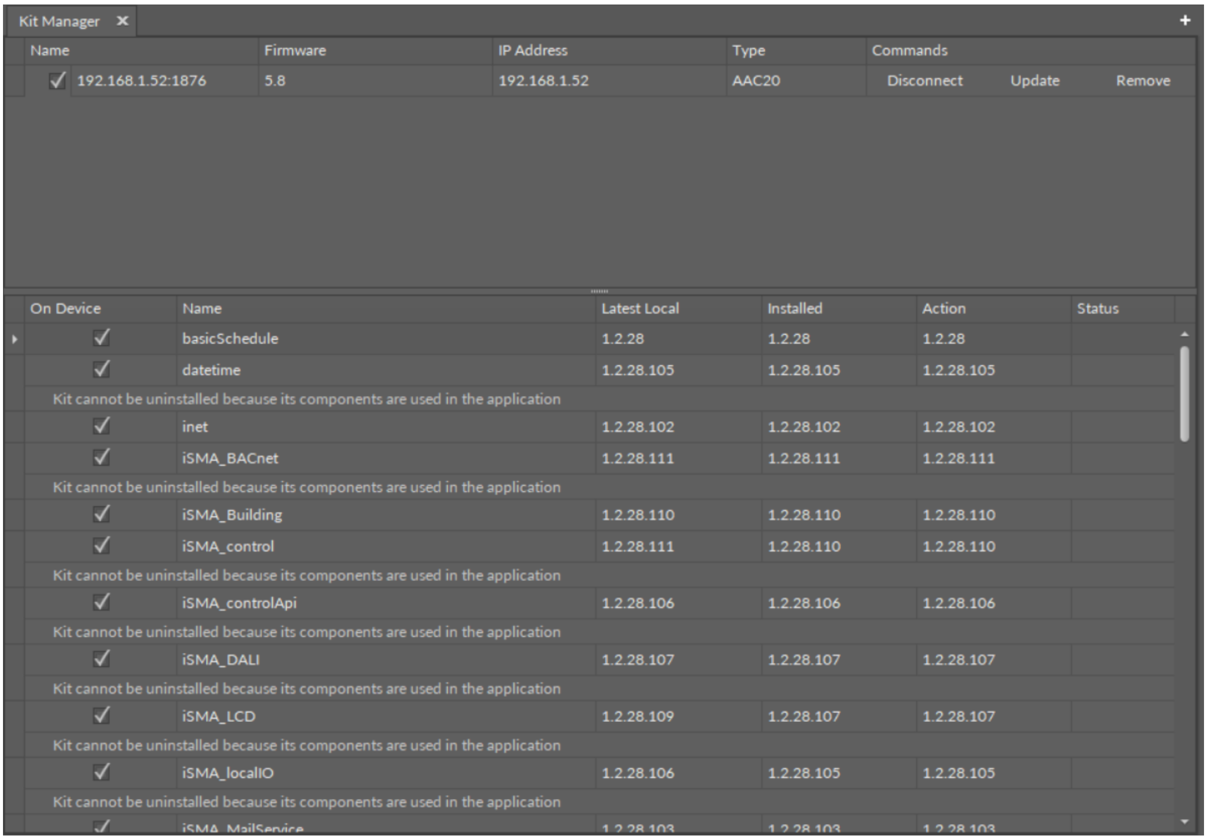
Attribute	Last Known Value
Licence status	Registered
Last connected	Now
Connection	Fast (average <1 ms)
Application name	demoApp
Serial number	22610091
Firmware	6.0
Type	AAC20
IP Address	192.168.1.52
Port	1876
Mac Address	54:10:EC:6B:FB:6A

Figure 85. Initiating the Kit Manager in the Object Properties window

After selecting the device in the Workspace Tree, its data will be shown in the Object Properties window, as shown on the figure above. It is possible that data will partially be not available due to a lack of information about the device until connecting with this device. Regardless of whether the device is connected or offline, the Kit Manager can be started by pressing the Kit Manager button, which will open with a device chosen.

7.5 Using the Kit Manager

After choosing the device to work with, using methods described earlier, a table with a list of kits is shown in the bottom part of the Kit Manager. In the upper part of the table, the kits, which are currently installed in the device, are grouped, and in the lower part of the table there are filtered kits, fitting to a selected device, which are available to be loaded into the device. After deselecting the device, the filter will be turned off, and a whole list of locally available kits will be shown in the iSMA Tool, see the figure below.



Name	Firmware	IP Address	Type	Commands		
✓ 192.168.1.52:1876	5.8	192.168.1.52	AAC20	Disconnect	Update	Remove

On Device	Name	Latest Local	Installed	Action	Status
✓	basicSchedule	1.2.28	1.2.28	1.2.28	
✓	datetime	1.2.28.105	1.2.28.105	1.2.28.105	
Kit cannot be uninstalled because its components are used in the application					
✓	inet	1.2.28.102	1.2.28.102	1.2.28.102	
✓	iSMA_BACnet	1.2.28.111	1.2.28.111	1.2.28.111	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_Building	1.2.28.110	1.2.28.110	1.2.28.110	
✓	iSMA_control	1.2.28.111	1.2.28.110	1.2.28.110	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_controlApi	1.2.28.106	1.2.28.106	1.2.28.106	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_DALI	1.2.28.107	1.2.28.107	1.2.28.107	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_LCD	1.2.28.109	1.2.28.107	1.2.28.107	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_localIO	1.2.28.106	1.2.28.105	1.2.28.105	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_MailService	1.2.28.103	1.2.28.103	1.2.28.103	

Figure 86. The Kit Manager

The columns in the above table are defined as follows:

- On Device: shows if a particular kit is already installed on the selected device;
- Name: the name of the kit;
- Latest Local: the number of the latest kit version installed locally in the iSMA Tool;
- Installed: the number of the kit version currently installed on the device;
- Action: the list of kits versions, which are available locally in the iSMA Tool and are ready to be installed on the device. A version number to be installed on the device is chosen from a drop-down menu (upgrade or downgrade process).
- Status: the column informs about an action that will be taken on the selected kit.

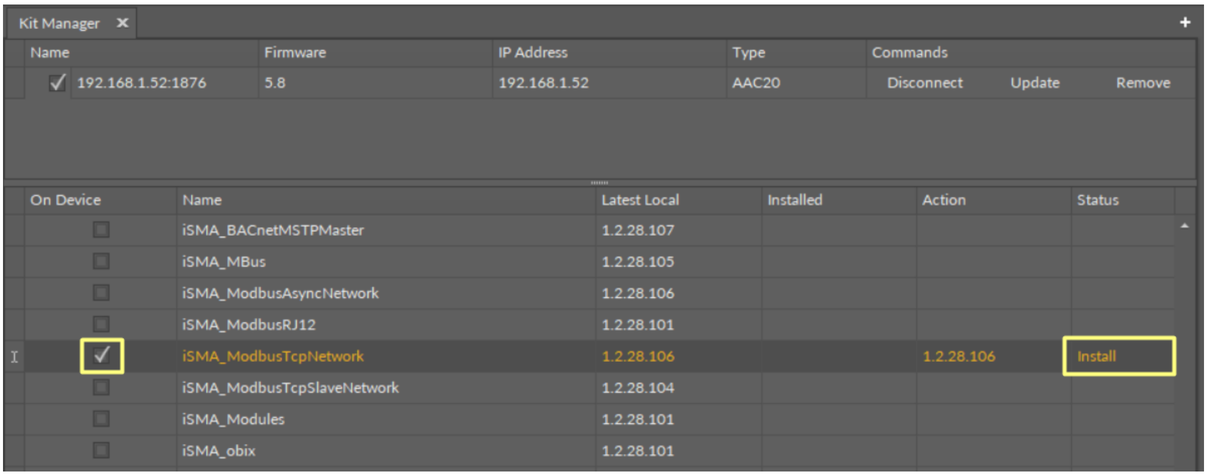
The user's role is limited to editing two columns: On Device and Action.

7.5.1 Operations on Kits

Below there are descriptions how to remove, replace, and add a kit to a device.

Adding Kits

In order to add the kit to the device, select its checkbox, and, if needed, define the version of added kit (by default, the newest version is selected). The Install command appears in the Status column, and the kit is added once the user confirms the Update command in the upper Device panel of the Kit Manager.



Kit Manager ✕

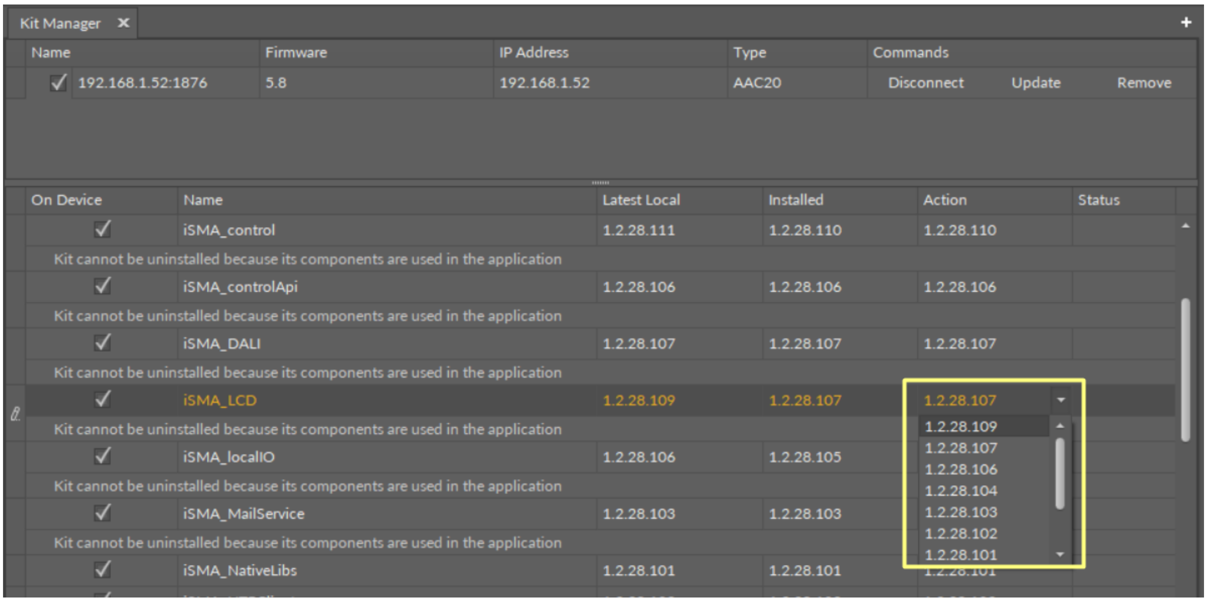
Name	Firmware	IP Address	Type	Commands		
✓ 192.168.1.52:1876	5.8	192.168.1.52	AAC20	Disconnect	Update	Remove

On Device	Name	Latest Local	Installed	Action	Status
□	iSMA_BACnetMSTPMaster	1.2.28.107			
□	iSMA_MBus	1.2.28.105			
□	iSMA_ModbusAsyncNetwork	1.2.28.106			
□	iSMA_ModbusRJ12	1.2.28.101			
✓	iSMA_ModbusTcpNetwork	1.2.28.106		1.2.28.106	Install
□	iSMA_ModbusTcpSlaveNetwork	1.2.28.104			
□	iSMA_Modules	1.2.28.101			
□	iSMA_obix	1.2.28.101			

Figure 87. Adding a kit

Upgrading/Downgrading Kits

The kit installed on the device may be replaced by installing its newer or older version. Both upgrade and downgrade operations are done by choosing a proper number of the kit version in the Action column.



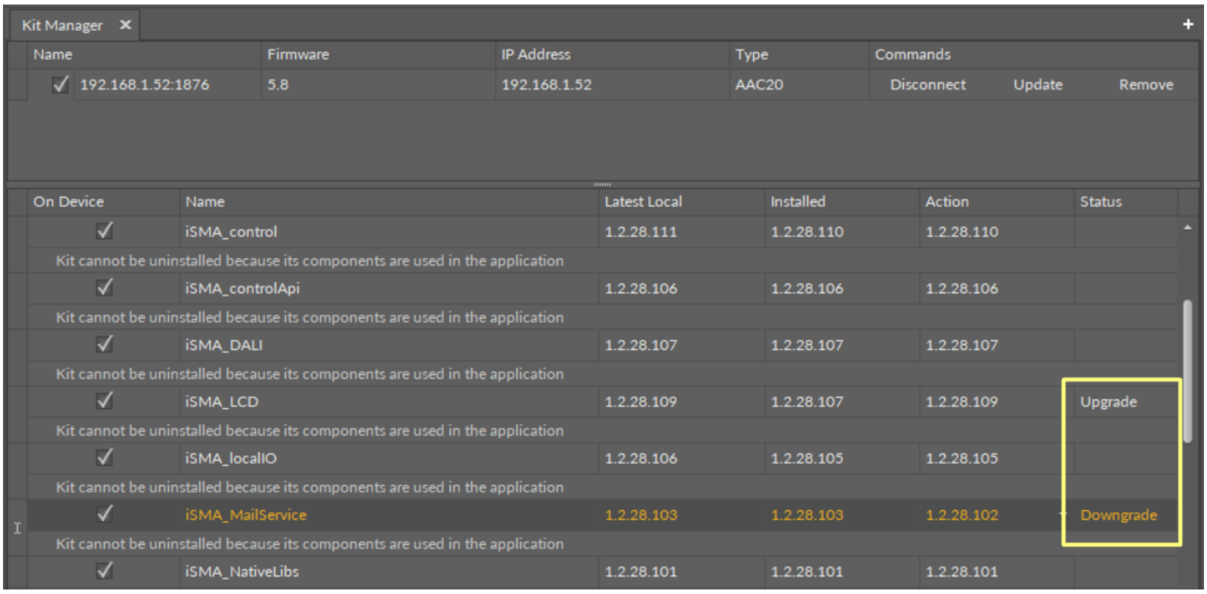
Kit Manager ✕

Name	Firmware	IP Address	Type	Commands		
✓ 192.168.1.52:1876	5.8	192.168.1.52	AAC20	Disconnect	Update	Remove

On Device	Name	Latest Local	Installed	Action	Status
✓	iSMA_control	1.2.28.111	1.2.28.110	1.2.28.110	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_controlApi	1.2.28.106	1.2.28.106	1.2.28.106	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_DALI	1.2.28.107	1.2.28.107	1.2.28.107	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_LCD	1.2.28.109	1.2.28.107	1.2.28.107 1.2.28.109 1.2.28.107 1.2.28.106 1.2.28.104 1.2.28.103 1.2.28.102 1.2.28.101	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_localIO	1.2.28.106	1.2.28.105		
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_MailService	1.2.28.103	1.2.28.103		
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_NativeLibs	1.2.28.101	1.2.28.101	1.2.28.101	

Figure 88. Selecting a version to upgrade or downgrade in the Action column

Depending if the current kit is replaced with a newer or older one, the iSMA Tool will display the planned action in the Status column, Upgrade or Downgrade.



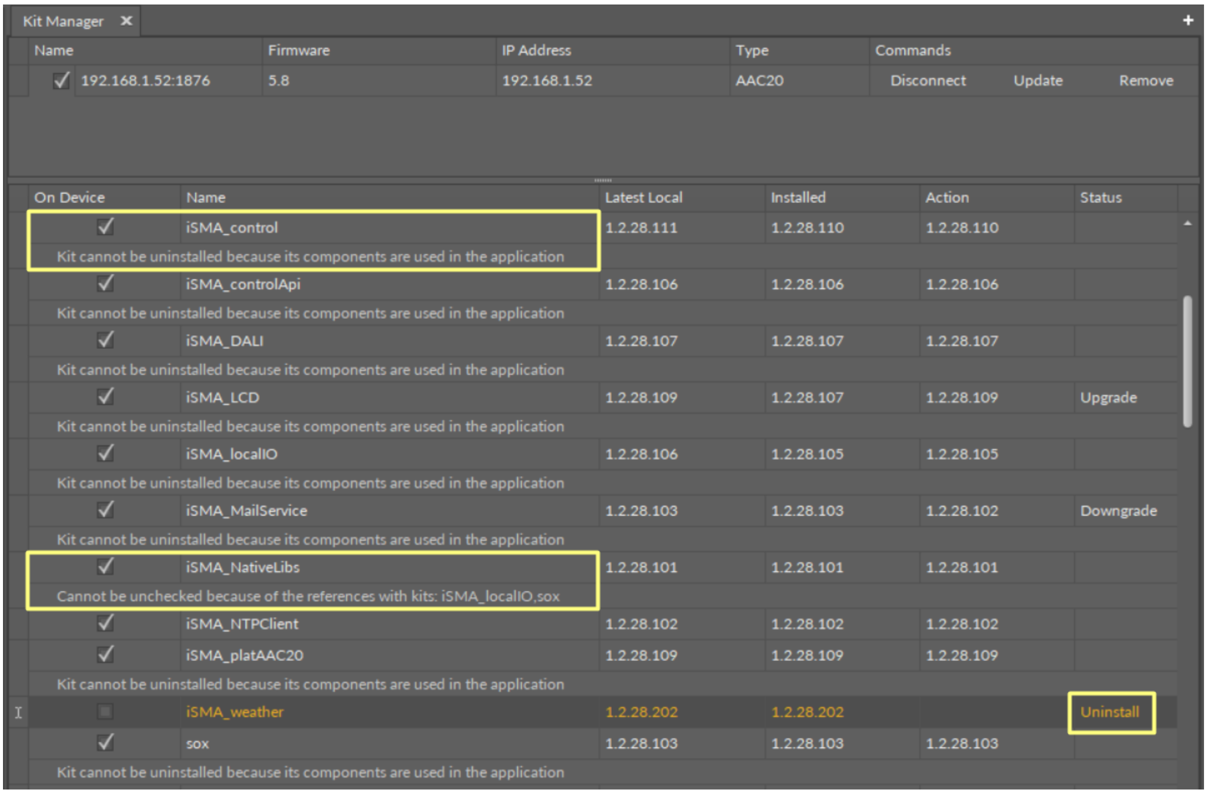
Name	Firmware	IP Address	Type	Commands		
✓ 192.168.1.52:1876	5.8	192.168.1.52	AAC20	Disconnect	Update	Remove

On Device	Name	Latest Local	Installed	Action	Status
✓	iSMA_control	1.2.28.111	1.2.28.110	1.2.28.110	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_controlApi	1.2.28.106	1.2.28.106	1.2.28.106	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_DALI	1.2.28.107	1.2.28.107	1.2.28.107	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_LCD	1.2.28.109	1.2.28.107	1.2.28.109	Upgrade
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_localIO	1.2.28.106	1.2.28.105	1.2.28.105	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_MailService	1.2.28.103	1.2.28.103	1.2.28.102	Downgrade
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_NativeLibs	1.2.28.101	1.2.28.101	1.2.28.101	

Figure 89. Upgrade or downgrade options

Removing/Uninstalling Kits

If a user deselects the checkbox, the selected kit is chosen to be removed. The Uninstall command appears in the Status column, and the kit is removed once the user confirms the Update command in the upper Device panel of the Kit Manager.



Name	Firmware	IP Address	Type	Commands		
✓ 192.168.1.52:1876	5.8	192.168.1.52	AAC20	Disconnect	Update	Remove

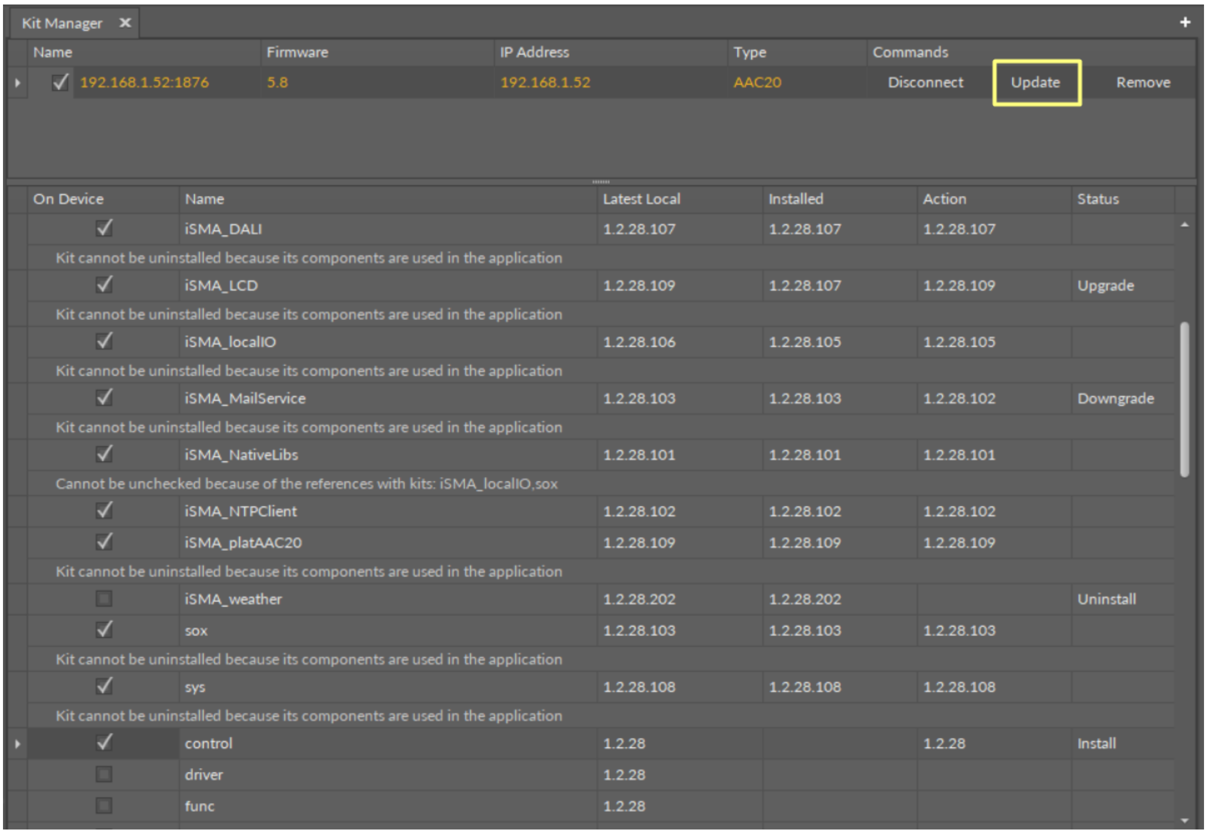
On Device	Name	Latest Local	Installed	Action	Status
✓	iSMA_control	1.2.28.111	1.2.28.110	1.2.28.110	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_controlApi	1.2.28.106	1.2.28.106	1.2.28.106	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_DALI	1.2.28.107	1.2.28.107	1.2.28.107	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_LCD	1.2.28.109	1.2.28.107	1.2.28.109	Upgrade
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_localIO	1.2.28.106	1.2.28.105	1.2.28.105	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_MailService	1.2.28.103	1.2.28.103	1.2.28.102	Downgrade
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_NativeLibs	1.2.28.101	1.2.28.101	1.2.28.101	
Cannot be unchecked because of the references with kits: iSMA_localIO,sox					
✓	iSMA_NTPClient	1.2.28.102	1.2.28.102	1.2.28.102	
✓	iSMA_platAAC20	1.2.28.109	1.2.28.109	1.2.28.109	
Kit cannot be uninstalled because its components are used in the application					
□	iSMA_weather	1.2.28.202	1.2.28.202		Uninstall
✓	sox	1.2.28.103	1.2.28.103	1.2.28.103	
Kit cannot be uninstalled because its components are used in the application					

Figure 90. Removing kits

In order for the kit to be removed, the components used in the application shall be removed, or the kit needs to be left installed.

7.5.2 Completing Operations - Updating Kits

After defining the kits to be added, removed, or changed the Update command needs to be confirmed in the upper Device panel of the Kit Manager. The Update command is displayed in the Commands column.



	Name	Firmware	IP Address	Type	Commands		
✓	192.168.1.52:1876	5.8	192.168.1.52	AAC20	Disconnect	Update	Remove

On Device	Name	Latest Local	Installed	Action	Status
✓	iSMA_DALI	1.2.28.107	1.2.28.107	1.2.28.107	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_LCD	1.2.28.109	1.2.28.107	1.2.28.109	Upgrade
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_localIO	1.2.28.106	1.2.28.105	1.2.28.105	
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_MailService	1.2.28.103	1.2.28.103	1.2.28.102	Downgrade
Kit cannot be uninstalled because its components are used in the application					
✓	iSMA_NativeLibs	1.2.28.101	1.2.28.101	1.2.28.101	
Cannot be unchecked because of the references with kits: iSMA_localIO,sox					
✓	iSMA_NTPClient	1.2.28.102	1.2.28.102	1.2.28.102	
✓	iSMA_platAAC20	1.2.28.109	1.2.28.109	1.2.28.109	
Kit cannot be uninstalled because its components are used in the application					
□	iSMA_weather	1.2.28.202	1.2.28.202		Uninstall
✓	sox	1.2.28.103	1.2.28.103	1.2.28.103	
Kit cannot be uninstalled because its components are used in the application					
✓	sys	1.2.28.108	1.2.28.108	1.2.28.108	
Kit cannot be uninstalled because its components are used in the application					
✓	control	1.2.28		1.2.28	Install
□	driver	1.2.28			
□	func	1.2.28			

Figure 91. Updating kits

While updating the kits in the device, the iSMA Tool interface is blocked for the time.

Warning!

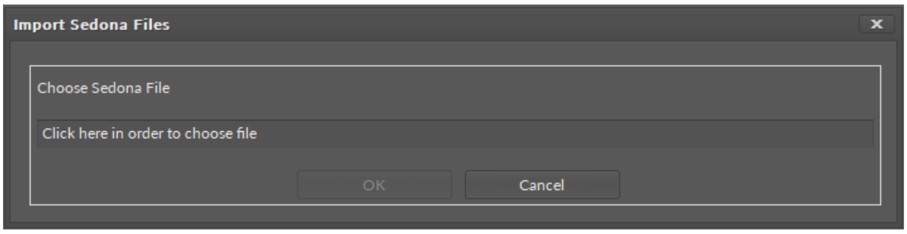
Finalizing the kits update requires restarting the device.

After restarting the device iSMA Tool reconnects with the device.

Note: The update process is monitored in detail in the Console window, which helps in analyzing the situation in case the update could not be completed.

7.6 Adding External Kits

The iSMA Tool allows to expand the default kit base with external kits, for example, user's own kits. To add them select the Import Sedona Files option, available in the main menu under Sedona. Once selected, a dialog window opens to specify the location of the imported files.



Import Sedona Files

Choose Sedona File

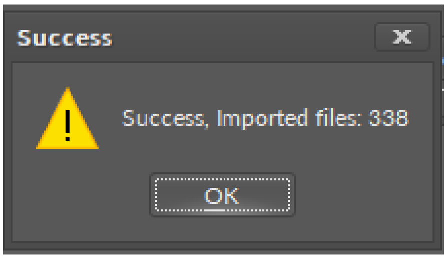
Click here in order to choose file

Figure 92. Adding external kits

The iSMA Tool may import single files or compressed folders with .zip extension containing:

a single kit; or whole packages containing kits, manifests, and .par files.

Note: The iSMA Tool can read the .zip files including other .zip files containing kits, manifests, and .par files, and can display a summary of how many files have been imported at the end of the process, in a window and in console, as shown on the figure below. If such package contains files, which are not kits, manifests, or .par files, they are skipped.



Success

Success, Imported files: 338

Figure 93. Successful upload

After a successful import the added kits are uploaded to the Kit Manager. The list of kits, which can be installed on the selected device, is displayed in the bottom part of the Kit Manager view It includes the kits, which were added manually. If the external kits do not appear on the list of available kits, it means they are not compatible with the selected device and cannot be uploaded. In this case, after deselecting the device, external kits will be displayed on the list in the lower part of the view, because in this mode the iSMA Tool shows all kits installed in it.

8 Application Manager

The Application Manager is a tool that allows to manage applications of a specific device and the applications saved locally in the iSMA Tool.

The Application Manager allows to:

- manage applications saved locally in the iSMA Tool;
- load applications to the device;
- download applications from the device.

Warning!

Loading an application to the device overwrites the application already saved in the device.

Warning!

Each loading of application to the device requires restarting the device.

Application Manager may be initiated in two ways: without defining the device it will work with and with defining the device it will work with.

8.1 Applications Installed in iSMA Tool

Opening the Application Manager without defining the device it will work with, allows to check what applications are currently saved in the iSMA Tool, which will be available to load into the device at the next stage.

To initiate the Application Manager with a list of applications available in the iSMA Tool go to the main menu: View -> Device Managers -> Application Manager.

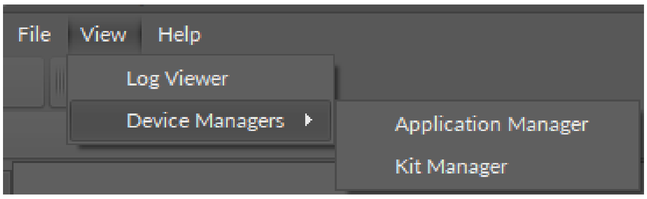
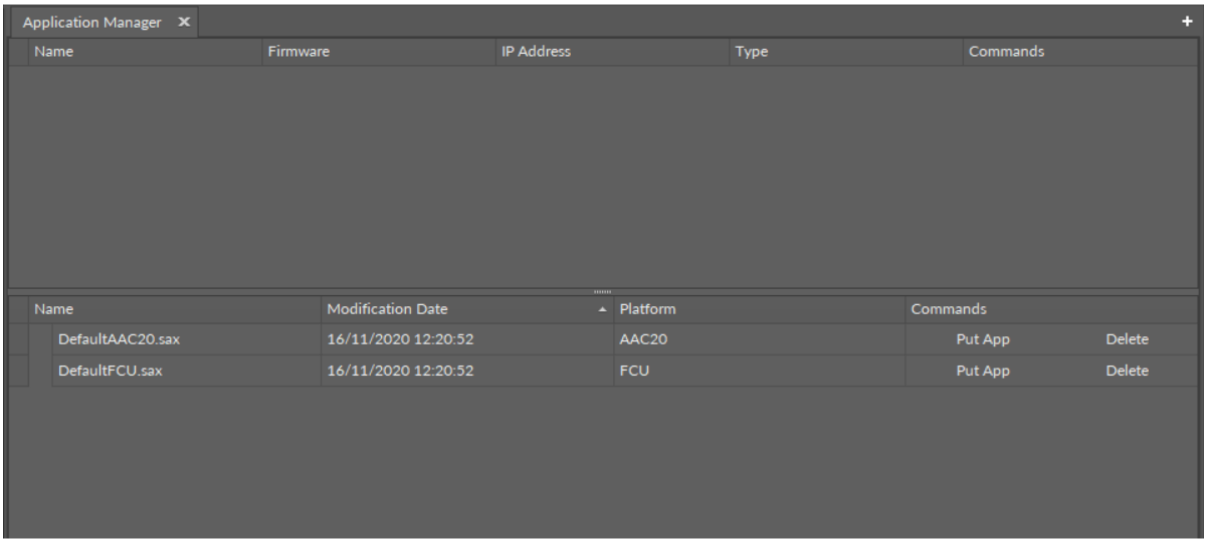


Figure 94. Accessing the Application Manager

The view which will appear is organized into two sections: upper, showing no connected devices, and lower with a list of available applications, see the figure below.



Application Manager ✕ +

Name	Firmware	IP Address	Type	Commands

Name	Modification Date	Platform	Commands	
DefaultAAC20.sax	16/11/2020 12:20:52	AAC20	Put App	Delete
DefaultFCU.sax	16/11/2020 12:20:52	FCU	Put App	Delete

Figure 95. Application Manager

Lower section shows a table where every row represents an application saved in the iSMA Tool along with a respective .sax file. In the iSMA Tool structures the .sax file is placed in the /home/Applications folder. Copying a custom application (.sax file) to this location allows to add it to Application Manager.

The meaning of columns is as follows:

- Name: the name of application and associated .sax file. The user can edit the name by clicking the application name. Renaming the application automatically changes the name of the .sax. file.
- Modification Date: the date of the last modification of the. sax file.
- Platform: the type of the device, which the application was downloaded from.
- Commands: buttons allowing to send the application to a particular device (Put App button) and the Delete button removing local application along with the .sax file from the iSMA Tool.

Warning!

Using the Delete button removes not only an application from the Application Manager list, but also a physical .sax file from the hard drive.

To download or upload an application from/to the device, the device to work with needs to be specifically defined.

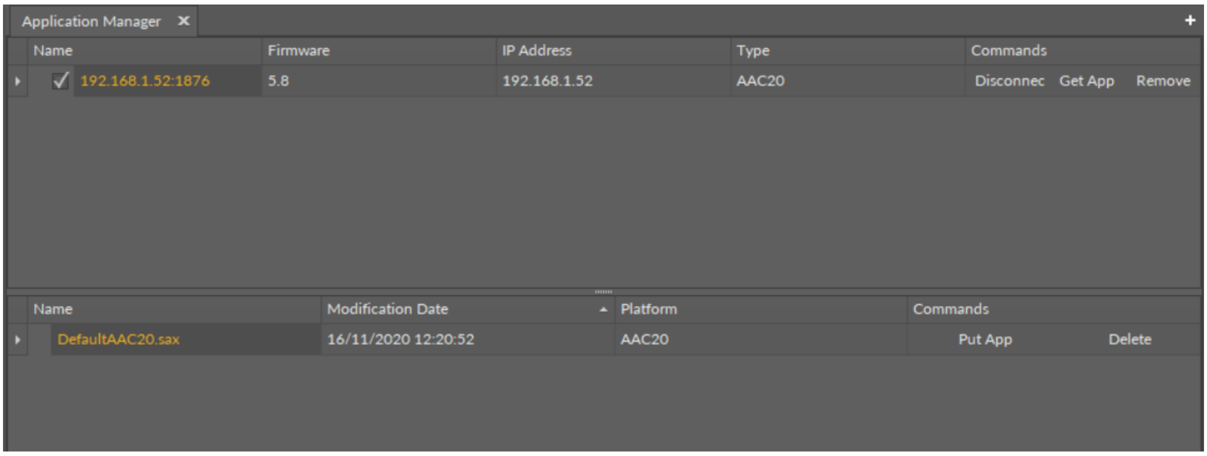
There are a few ways to initiate the Application Manager:

- initiating the Application Manager from the context menu;
- initiating the Application Manager from the Object Properties window.

The rules of initiating the iSMA Tool Managers, as well as choosing the device to work with, are discussed in detail in the Kit Manager section.

8.2 Using the Application Manager

After defining the device to work with, the upper part of Application Manager will look similarly to the figure below. In the upper part there is a list of devices, and only one device can by chosen at a time by selecting a checkbox.



Application Manager

Name	Firmware	IP Address	Type	Commands
✓ 192.168.1.52:1876	5.8	192.168.1.52	AAC20	Disconnec Get App Remove

Name	Modification Date	Platform	Commands
DefaultAAC20.sax	16/11/2020 12:20:52	AAC20	Put App Delete

Figure 96. Application Manager

The columns in the above table are defined as follows:

- On Device: shows if a particular kit is already installed on the selected device;
- Name: the name of the kit;
- Latest Local: the number of the latest kit version installed locally in the iSMA Tool;
- Installed: the number of the kit version currently installed on the device;
- Action: the list of kits versions, which are available locally in the iSMA Tool and are ready to be installed on the device. A version number to be installed on the device is chosen from a drop-down menu (upgrade or downgrade process).
- Status: the column informs about an action that will be taken on the selected kit.

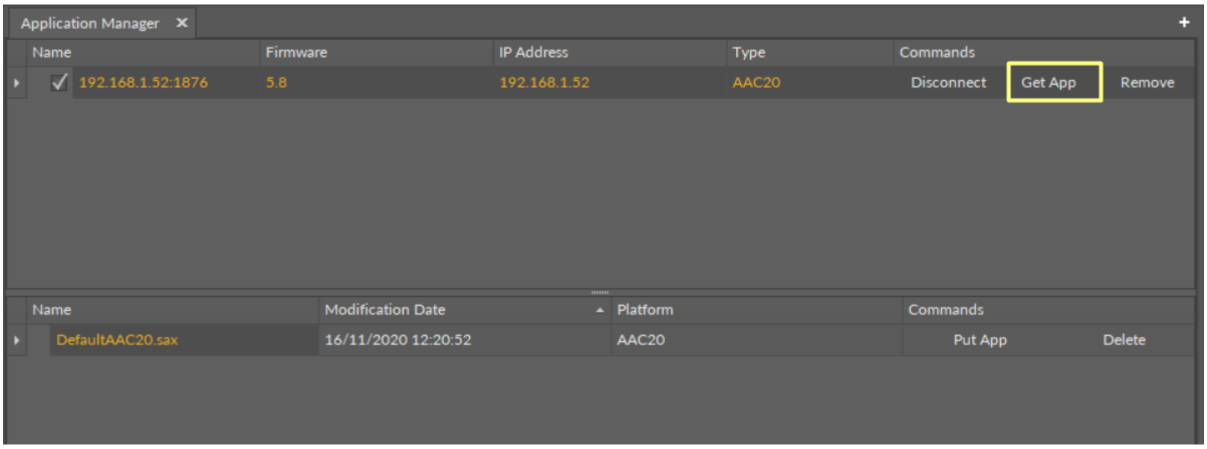
The user's role is limited to editing two columns: On Device and Action.

8.2.1 Operations on Applications

Below there are descriptions how to remove, replace, and add a kit to a device.

Downloading Application from Device (Get App)

After connecting with a chosen device its application can be downloaded by pressing the Get App button, see the figure below.



Application Manager

Name	Firmware	IP Address	Type	Commands
✓ 192.168.1.52:1876	5.8	192.168.1.52	AAC20	Disconnect Get App Remove

Name	Modification Date	Platform	Commands
DefaultAAC20.sax	16/11/2020 12:20:52	AAC20	Put App Delete

Figure 97. Download application

After initiating the command, a starting process reads an application from the device and creates the .sax file (with a name consisting of the application name and the device's IP address). This process blocks the iSMA Tool for the time of its operation. The newly created file will be added to the list of available applications saved in the iSMA Tool, and the physical .sax file will be located in the iSMA Tool main folder (/home/Applications). The default name of the application file may be changed by editing the Name column.

Note: Downloading an application from the device does not stop or restart the device. Downloading does not disrupt the device's work in any way other than pausing its interface for the time of downloading.

Note: Application downloading process is monitored in detail in the Console window, which helps to analyze the situation, in case downloading the application could not be completed.

Uploading Application to Device (Put App)

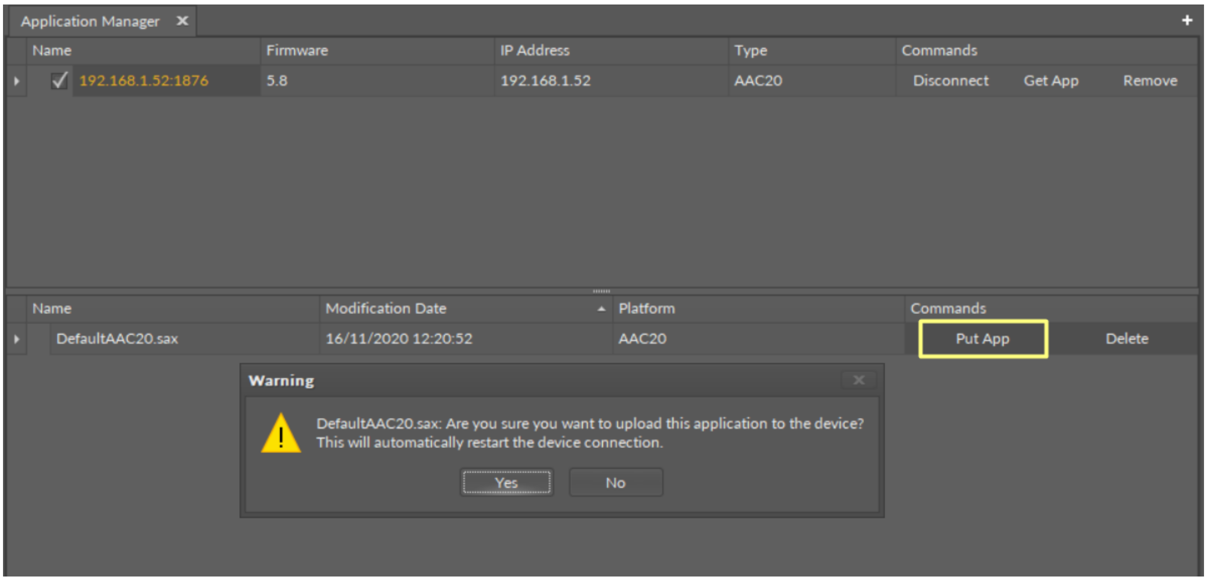
Application uploading procedure needs to begin with defining the device to work with Application Manager by selecting a checkbox for a particular device in the upper part of the Manager view.

Next, one of the available applications in the lower part of the Manager needs to be selected by pressing the Put App button in the Commands column.

Warning!

Pressing the Put App button will overwrite the application already installed in the device and reset the device in the final phase of the process.

The figure below shows a situation before beginning of the application upload into the iSMA-B-AAC20 controller, where the user is asked for confirmation due to overwriting of the application currently installed in controller.



Name	Firmware	IP Address	Type	Commands
✓ 192.168.1.52:1876	5.8	192.168.1.52	AAC20	Disconnect Get App Remove

Name	Modification Date	Platform	Commands
DefaultAAC20.sax	16/11/2020 12:20:52	AAC20	Put App Delete

Warning

DefaultAAC20.sax: Are you sure you want to upload this application to the device? This will automatically restart the device connection.

Figure 98. Put App notice

After approving the confirmation, a process blocking the iSMA Tool interface runs for a period of time, when the iSMA Tool uploads a chosen application and restarts and reconnects the device.

Note: Application uploading process is monitored in detail in the Console window, which helps to analyze the situation when application upload could not finish.

9 Standard Views

The iSMA Tool defines few standard views available after choosing the component. They are: the Wire Sheet, Property Sheet, and Slot Sheet. These views are displayed in the iSMA Tool Main Window on separate tabs. Despite the standard views mentioned above, available for any component, there can also be other, dedicated views for particular components.

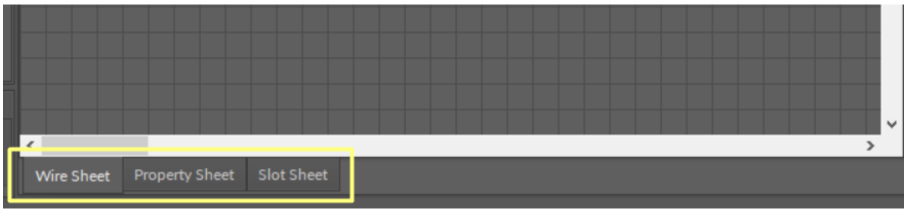


Figure 99. Component's standard views

9.1 Common Actions

Standard views for components have different roles and functions, however, they share common actions, which facilitate working with components.

9.1.1 View History

While working on application, the user switches between views to present objects being worked on in a most convenient way. Double clicking on a component in the Workspace Tree opens a corresponding default view. As a convenience matter, the iSMA Tool offers a view navigation history to allow the user to return to previously opened tabs.

The view navigation history is available on a toolbar under the main menu. There are three navigation history buttons in the toolbar: Previous, Next, and History.

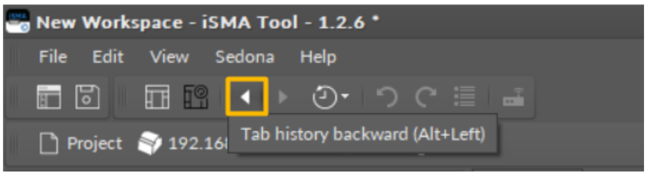

New Workspace - iSMA Tool - 1.2.6 *
File
Edit
View
Sedona
Help
Project
192.16
Tab history backward (Alt+Left)

Figure 100. The Tab backward button

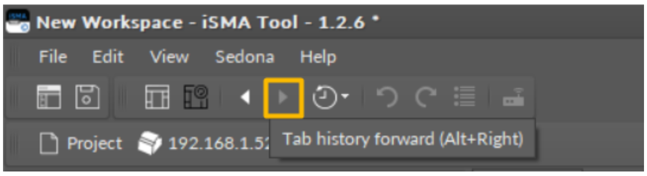

New Workspace - iSMA Tool - 1.2.6 *
File
Edit
View
Sedona
Help
Project
192.168.1.5
Tab history forward (Alt+Right)

Figure 101. The Tab forward button

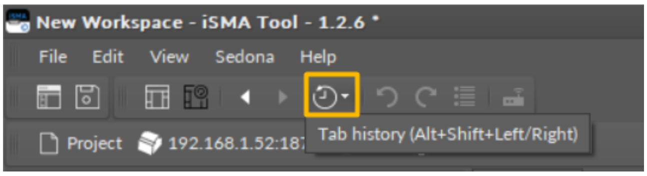

New Workspace - iSMA Tool - 1.2.6 *
File
Edit
View
Sedona
Help
Project
192.168.1.52:18
Tab history (Alt+Shift+Left/Right)

Figure 102. The Tab history button

These actions can also be invoked using keyboard shortcuts: Alt + Left, Alt + Right, or Alt + Shift + Left/Right key combinations.

The view history is presented depending on how it was invoked. Clicking a toolbar button presents a dropdown list:

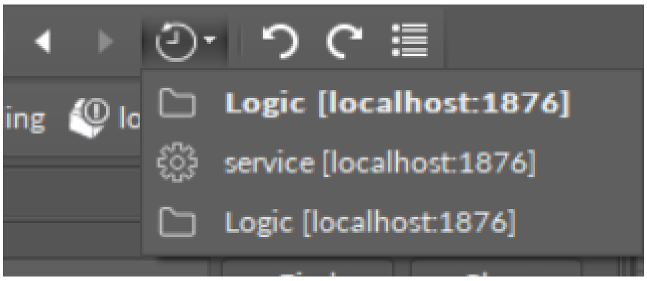


Figure 103. Tabs history

Pressing an Alt + Shift + Left/Right keyboard shortcut presents a flyout panel:

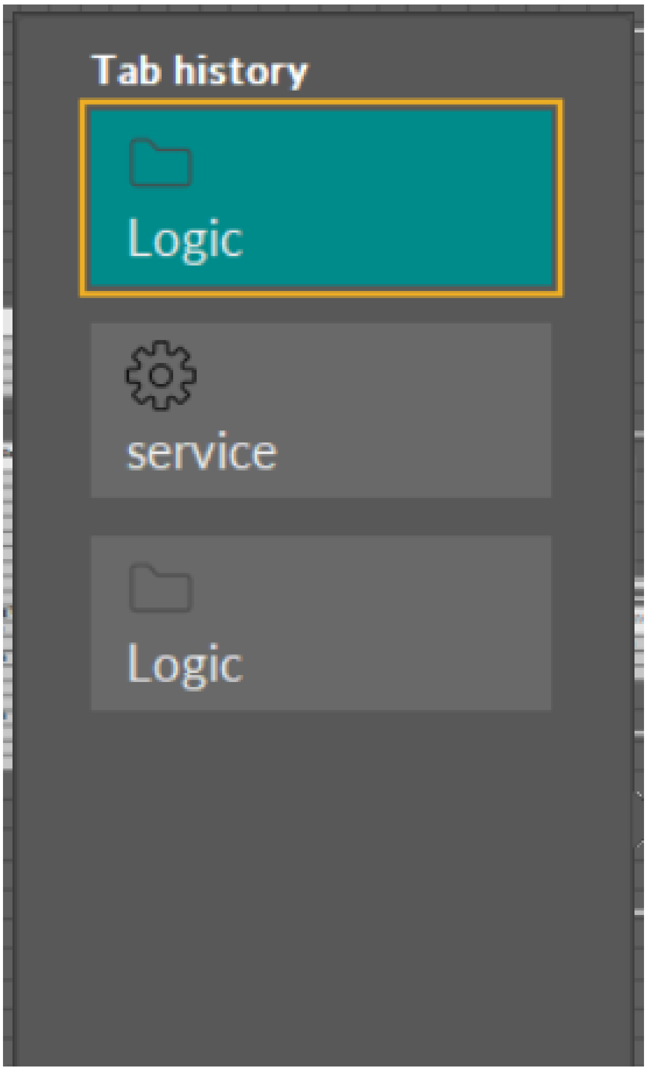


Figure 104. Tabs history flyout panel

9.1.2 Undo / Redo Actions

In case of an unintended action (for example, a component deleted by mistake, erroneously added link, randomly moved components), the iSMA Tool offers undo/redo actions.

With standard key combinations of Ctrl+Z and Ctrl+Y, the iSMA Tool allows to undo/redo actions step-by-step.

Pressing either key combination brings an undo/redo history flyout panel to help identify what will be undone/redone next. The panel stays in place while the user keeps holding the Ctrl button down so it is visible what is available to be undone/redone next if multiple changes are needed.

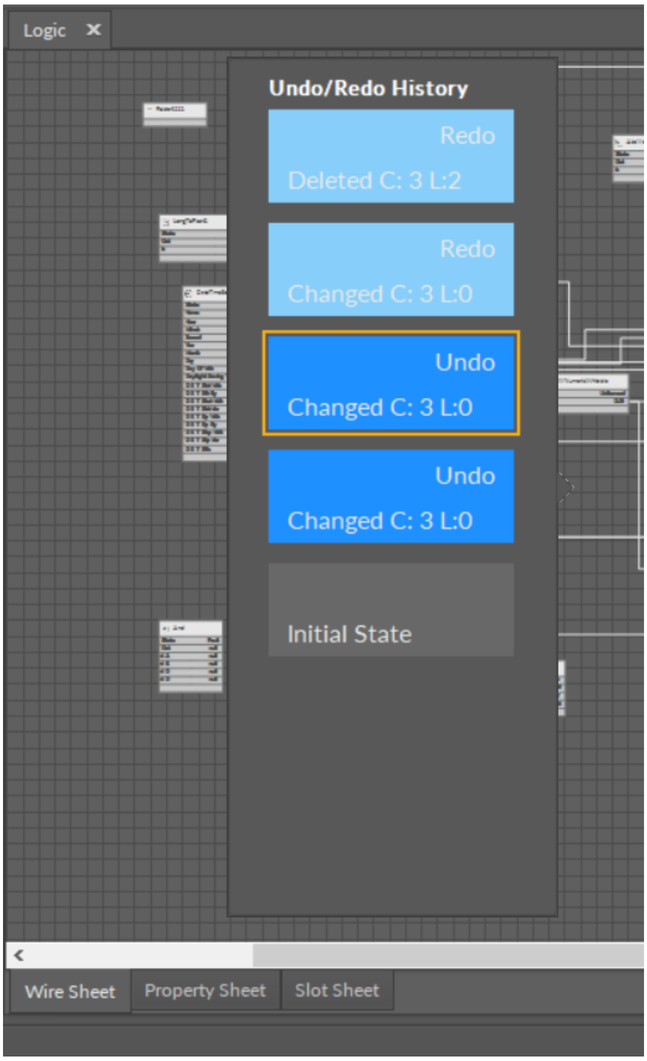


Figure 105. Undo/redo action flyout panel

It is also possible to invoke undo/redo actions or open the flyout panel clicking corresponding buttons on the toolbar:

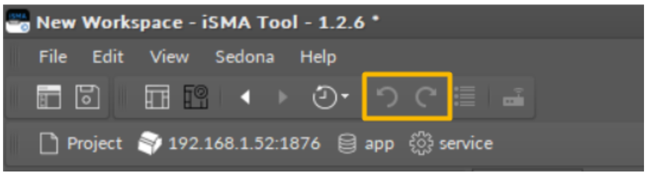


Figure 106. Undo/redo button

Warning!

As many operations on components involve immediate changes to the device over the wire, and the device itself could be modified by another user of iSMA Tool software remotely, it is possible that some actions will fail to complete. To minimize risks of loses to the application the user is creating, the undo/redo history is cleared in case of unrecoverable communication issues.

9.1.3 Reconnecting Devices

When working with multiple devices, it is often a case that multiple tabs are opened but devices (the tabs refer to) are disconnected. It is possible to reconnect them all at once by clicking a single button on the toolbar.

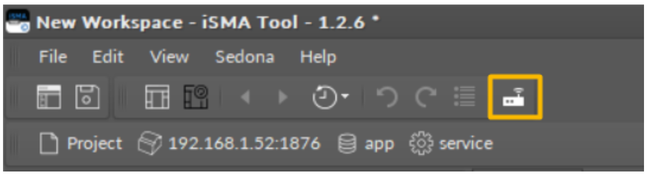
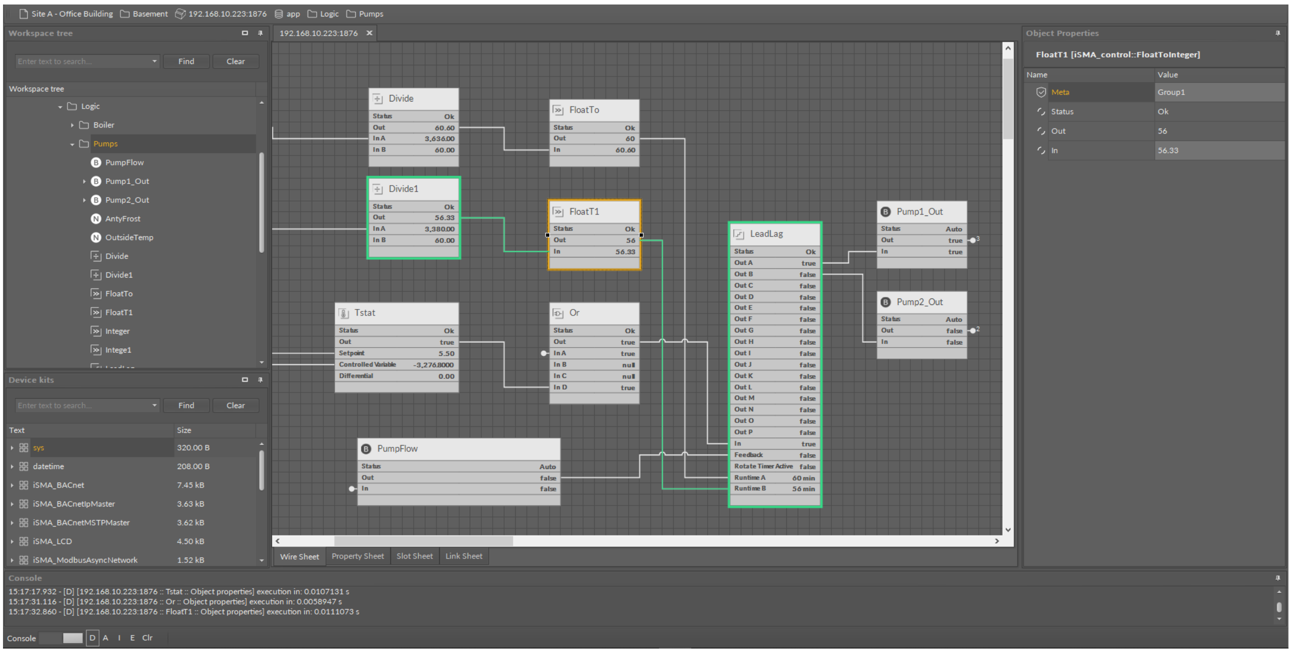


Figure 107. The Reconnect button

Once the action is invoked, all devices with at least one tab open are reconnected. To avoid swamping network communication, actual connection commands are sent once the user switches to a corresponding tab.

9.2 Wire Sheet

The Wire Sheet is a graphic view which shows all the inferior components (along with links) for the selected superior component (double-clicked in the Workspace Tree). Each component in the Wire Sheet view is shown as a rectangle with its name and icon placed in a header, and slot names and values in rows underneath. The link is displayed as a line connecting two component slots, between which values are transferred. The figure below shows the Wire Sheet view with corresponding list of components in the Workspace Tree for the Application service.



FloatT1 [iSMA_control::FloatToInteger]

Name	Value
Meta	Group1
Status	Ok
Out	56
In	56.33

```
15:17:17.932 - [D] [192.168.10.223:1876 :: Tstat :: Object properties] execution in: 0.0107131 s
15:17:31.116 - [D] [192.168.10.223:1876 :: Or :: Object properties] execution in: 0.0058947 s
15:17:32.860 - [D] [192.168.10.223:1876 :: FloatT1 :: Object properties] execution in: 0.0111073 s
```

Figure 108. The Wire Sheet along with the Workspace Tree

In the Wire Sheet view the user can define the location of components by dragging them by the header. To make moving components easier the Wire Sheet displays a grid allowing precise placement of the component. The grid makes organizing of components and their alignment easier.

9.2.1 Components and Links Representation

Every component in the Wire Sheet view has a following look and behavior:

- Components have headers with icons and components' names.
- Under the component's header every row represents a single slot with a name, value, and unit.
- The Meta slot is not displayed.
- Slots hidden by default are not displayed.
- Component may be extended left and right between 5 and 32 grid sections. This allows the component to be adjusted optimally, so all the slots names and their values can be fully visible.
- Component can be moved on the Wire Sheet into any place by clicking and holding a component's header or a middle part of any slot. Releasing the component defines its new location, after which links between this and other components may be renewed.
- Moving of component can be cancelled by pressing an Esc button on the keyboard.
- Component can also be moved by pressing an Alt key and clicking any part of component. This allows to grab and drag the component also behind the place, where link assigning begins.
- Selected (clicked) component is marked with an orange frame, and is treated by the iSMA Tool as an object chosen by user, contrary to the component surrounded by a green frame, which is treated by the iSMA Tool as a reference (auxiliary) object, not selected.
- A context menu available at right-click provides functions, which can be performed on a component, regardless of the view it is displayed in.

Every link in the Wire Sheet view has a following look and behavior:

- Links between components reflect the connections between specific slots of connected components.
- Link guarantees sending the value of source component slot to a target component slot.
- Single link cannot branch out and is always unidirectional from the output of the source component slot to the input of target component slot.
- Only one output link can be defined for an input (left) slot.
- More than one output links may come out of one output (right) slot.
- Links connecting components from different Wire Sheet are graphically represented as short segments coming in/out of slots ended with a full circle.
- A context menu available under right-click allows to remove a selected link.
- The user has no influence on a link route, i.e. the way it will be drawn, since the process is done automatically.

9.2.2 Navigation

Navigation on the Wire Sheet view is carried out by zooming in/out and moving the object in the view.

Zooming the view in or out requires turning a mouse wheel upward or downward. It is worth noting that zooming is centered where the mouse pointer is. In order to zoom in or out a particular Wire Sheet fragment, the mouse pointer needs to be moved into this area, so it does not escape the field of view.

In order to scroll the Wire Sheet view, move the sliders at the bottom and right side of the window or press the Ctrl button and click and move up or down.

After opening the Wire Sheet view, the iSMA Tool automatically adjusts the zoom allowing all components to be visible.

Pressing the F key at any time will automatically rescale and recenter the view, so all the components can be visible.

9.2.3 Selecting and Marking of Components and Links

The Wire Sheet view has been designed to make it easy for the user to analyze the logic of the program by proper marking of the inferior components and links in relation to a selected component or link, see the figure below.

Selecting and Marking of Components

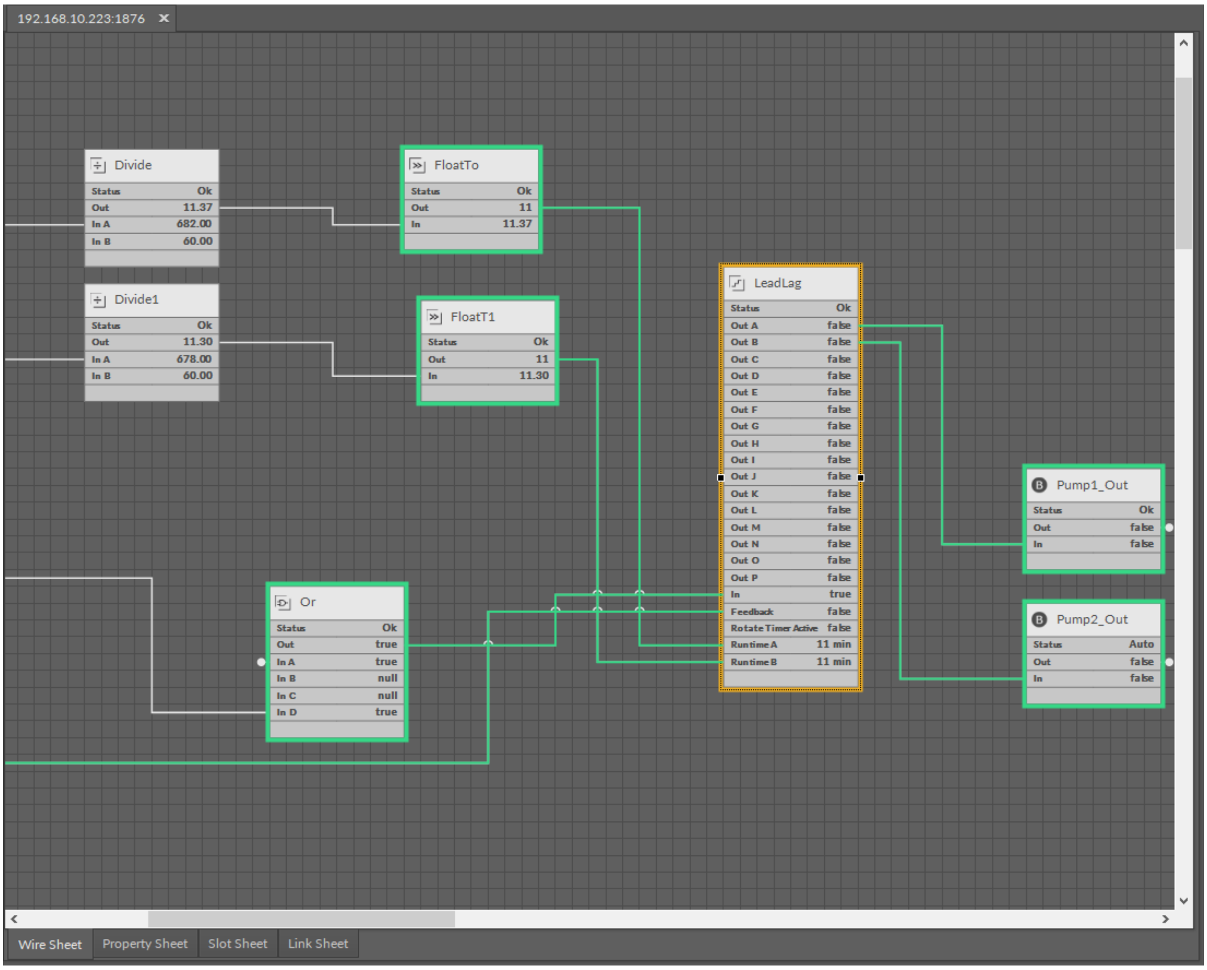


Figure 109. The GreaterOrEqual component selected

A LeadLag component has been selected on the figure and is marked with an orange frame. If it will be removed or moved, or duplicated, it has an impact on the links connected to it and their components. The whole path of connected objects has been highlighted in green. A coloring mechanism shown above may be very useful to analyze complex programs with many connections, links and components.

Selecting and Marking of Links

Similar color highlighting takes place if a link is selected (marked in orange). In such case the iSMA Tool will automatically mark with a green color all components connected with this link. It allows the user to be aware which components is the link connected to, and which will be affected in case it is, for example, deleted (see the figure below).

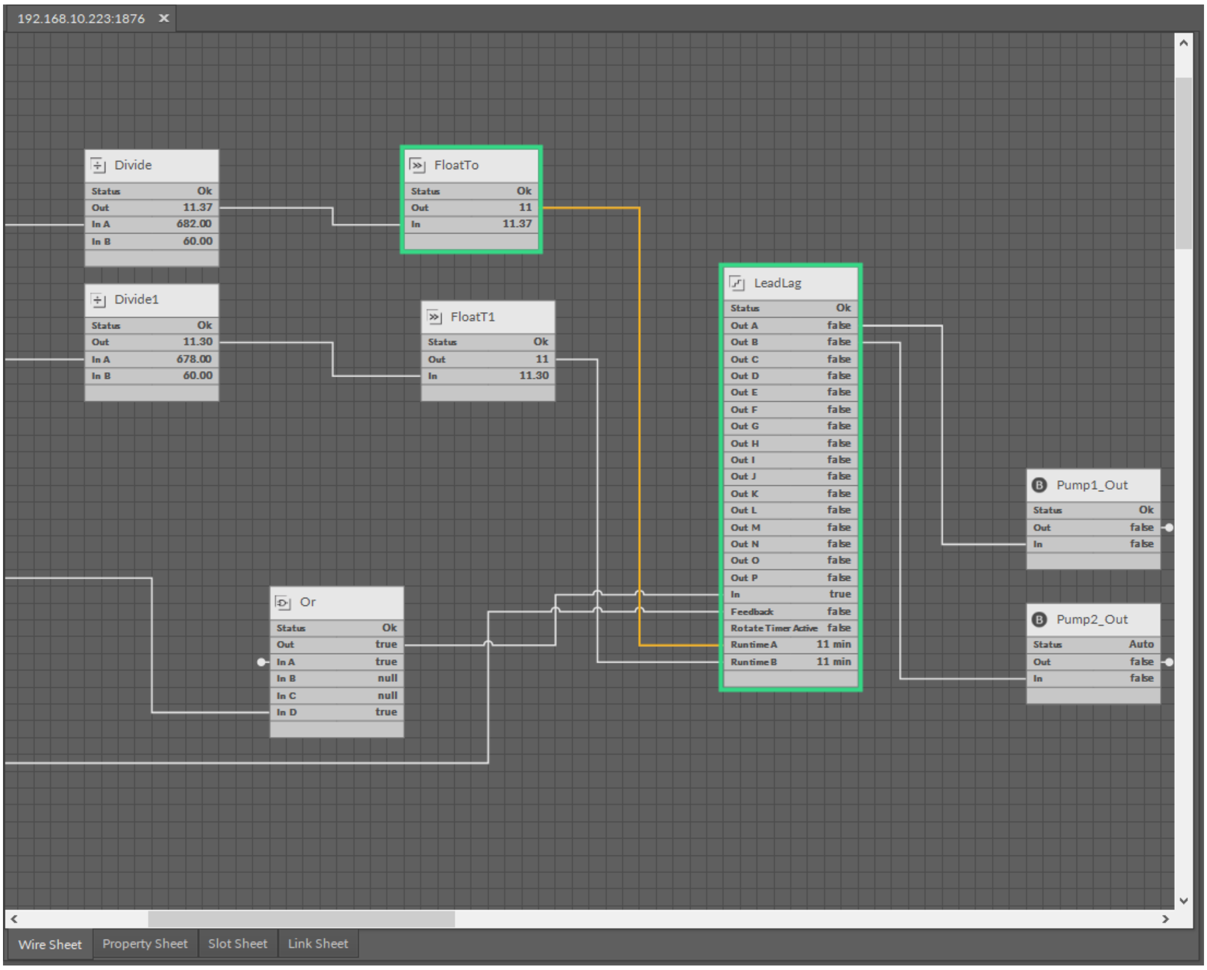


Figure 110. Selected links

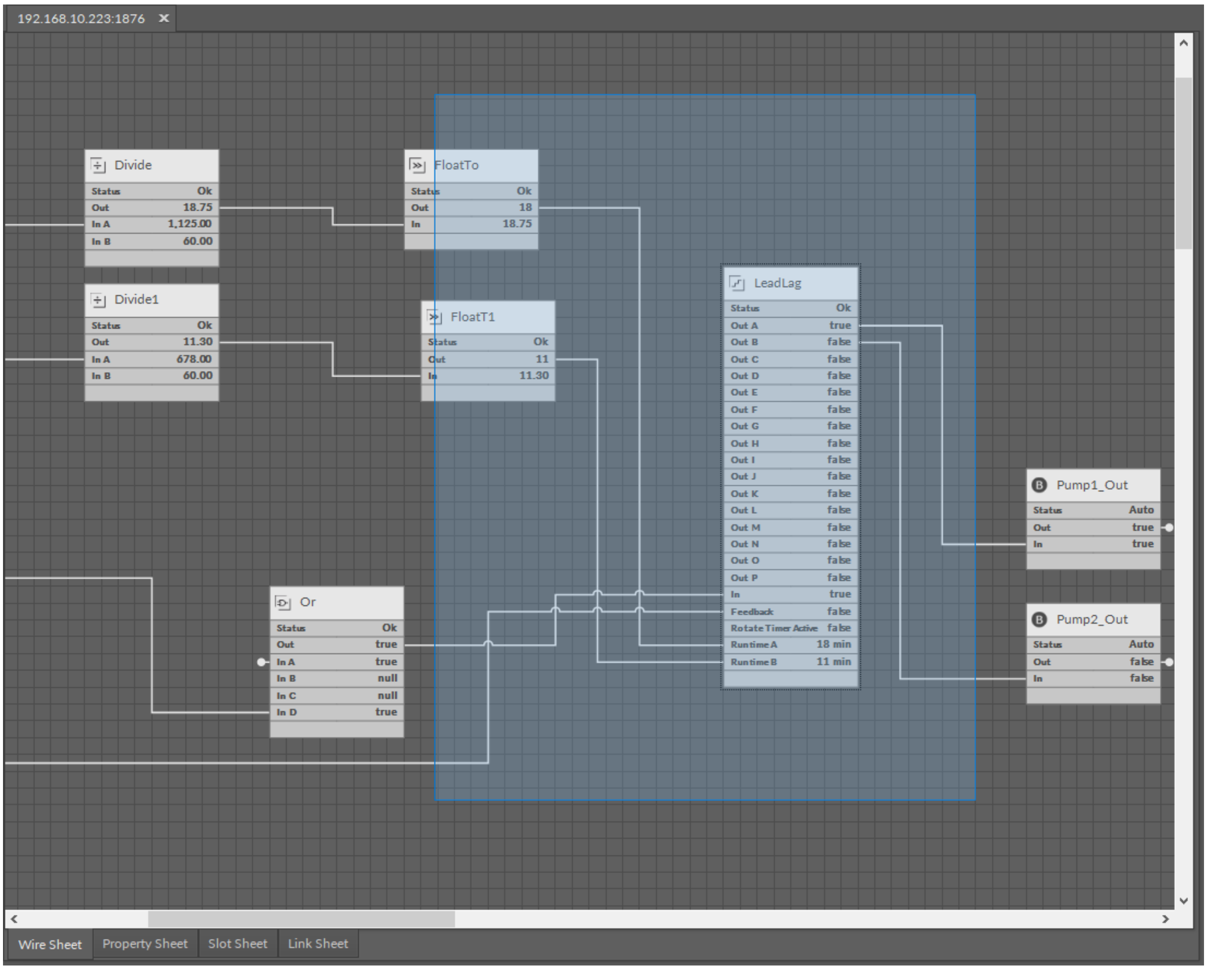
Window and Crossing Selection

Apart from selecting elements with Ctrl key, in the Wire Sheet view there is also another method of selecting components and links. The method involves defining a rectangular area which marks objects to be selected. It can either be a window selection or a cross selection.

Both types require marking a square area in the Wire Sheet view by clicking in an empty area and dragging a rectangle over the objects to be selected. Release the rectangle when all the selected objects are included.

Window Selection

In case of a window selection (see the figures below), a rectangle dragged from left to right is blue. Only objects fully covered by the rectangle are selected (orange color), here these are the LeadLag component and its two links. The remaining elements are not selected because they were not fully covered by the blue rectangle. The elements dependent on the selected LeadLag component and selected links are marked in green.



Divide
Status Ok
Out 18.75
In A 1,125.00
In B 60.00

Divide1
Status Ok
Out 11.30
In A 678.00
In B 60.00

FloatTo
Status Ok
Out 18
In 18.75

FloatT1
Status Ok
Out 11
In 11.30

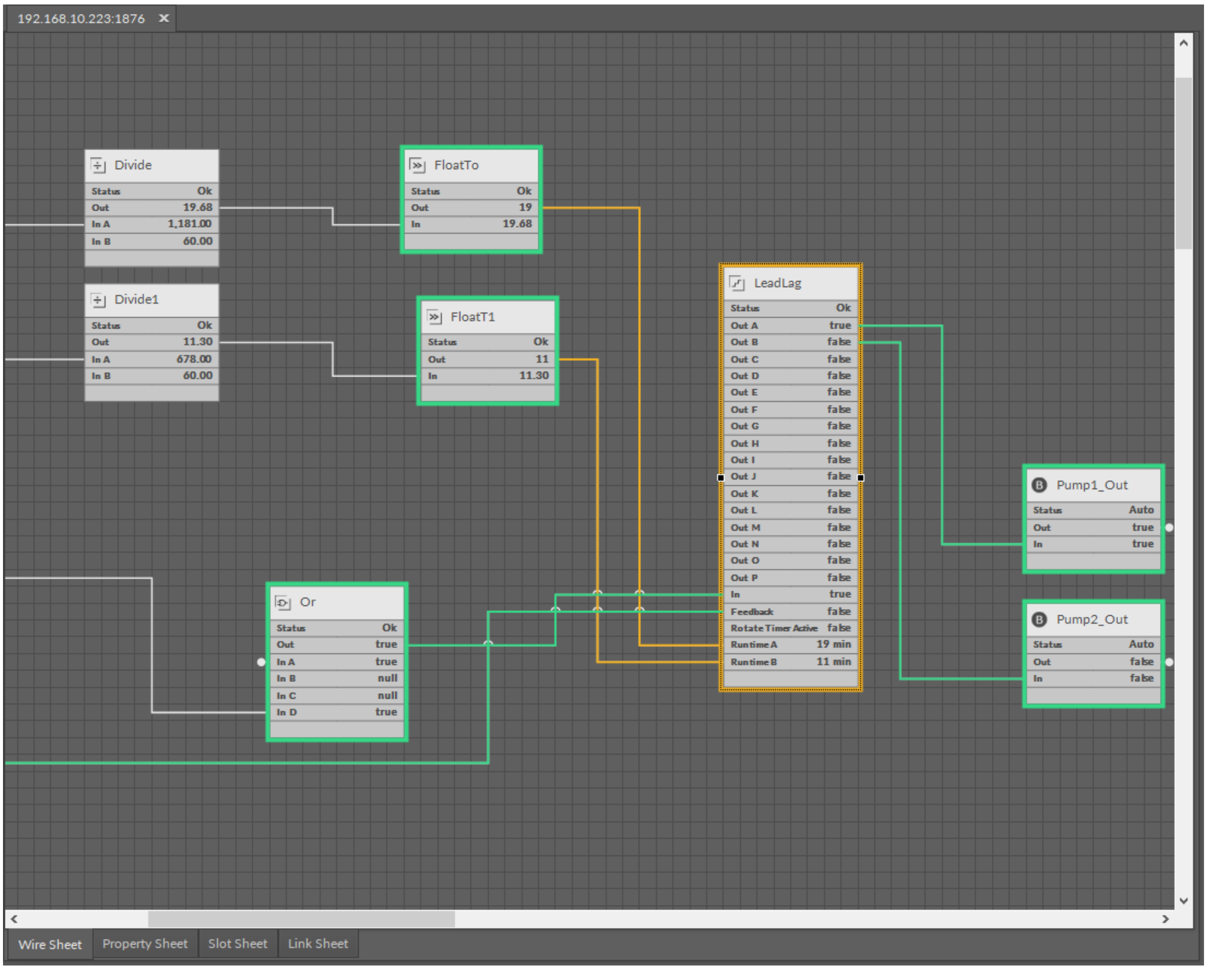
LeadLag
Status Ok
Out A true
Out B false
Out C false
Out D false
Out E false
Out F false
Out G false
Out H false
Out I false
Out J false
Out K false
Out L false
Out M false
Out N false
Out O false
Out P false
In true
Feedback false
Rotate Timer Active false
Runtime A 18 min
Runtime B 11 min

Or
Status Ok
Out true
In A true
In B null
In C null
In D true

Pump1_Out
Status Auto
Out true
In true

Pump2_Out
Status Auto
Out false
In false

Figure 111. The window selection



Divide
Status Ok
Out 19.68
In A 1,181.00
In B 60.00

Divide1
Status Ok
Out 11.30
In A 678.00
In B 60.00

FloatTo
Status Ok
Out 19
In 19.68

FloatT1
Status Ok
Out 11
In 11.30

LeadLag
Status Ok
Out A true
Out B false
Out C false
Out D false
Out E false
Out F false
Out G false
Out H false
Out I false
Out J false
Out K false
Out L false
Out M false
Out N false
Out O false
Out P false
In true
Feedback false
Rotate Timer Active false
Runtime A 19 min
Runtime B 11 min

Or
Status Ok
Out true
In A true
In B null
In C null
In D true

Pump1_Out
Status Auto
Out true
In true

Pump2_Out
Status Auto
Out false
In false

Figure 112. The window selection

Crossing Selection

In case of a cross selection (see the figures below) the rectangle is dragged the opposite way, **from right to left**, and it is red. All the objects, even partially included in the rectangle, are selected (orange color), here these are the LeadLag component along with its links and components FloatTo and FloatT1 with output links.

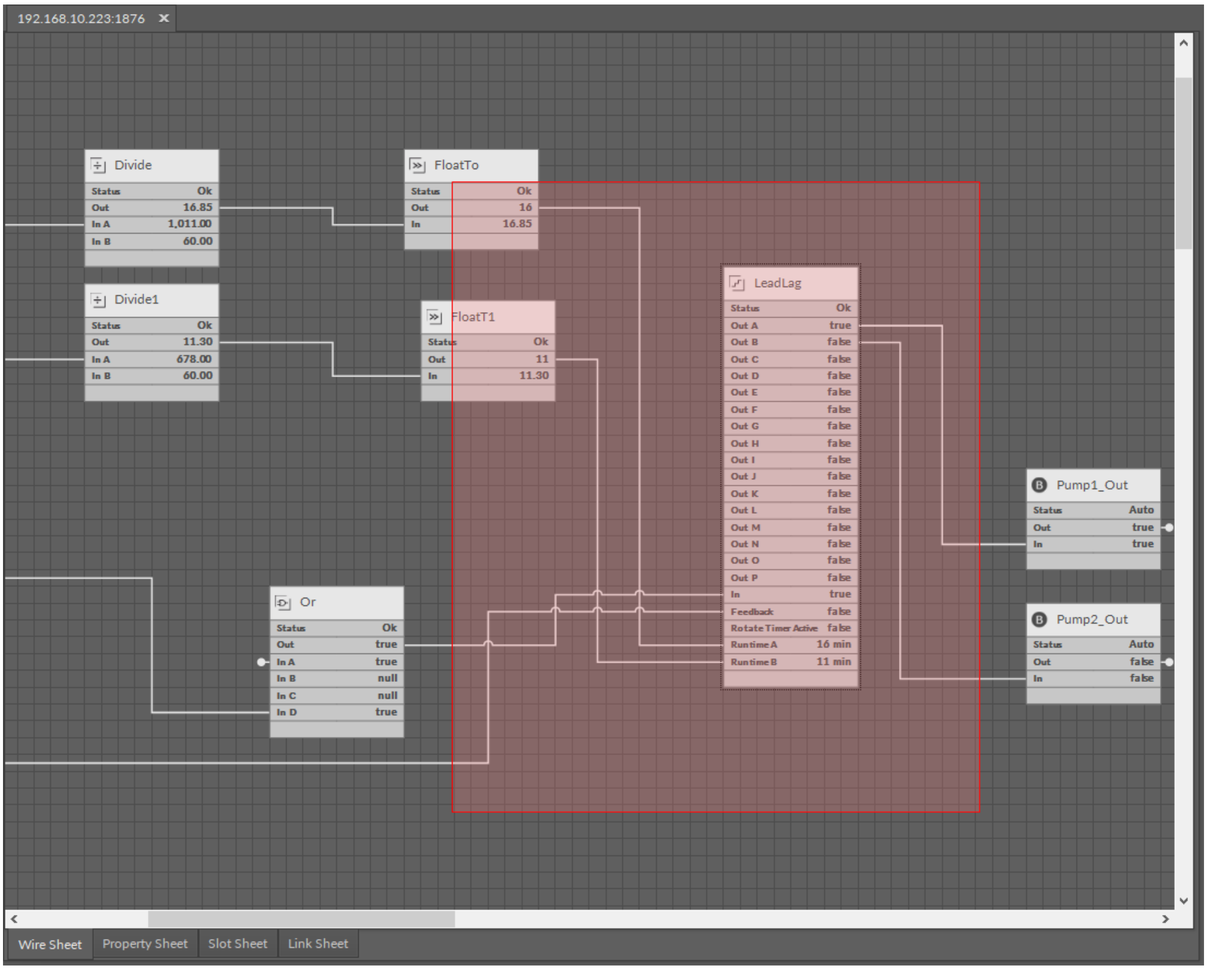


Figure 113. The crossing selection

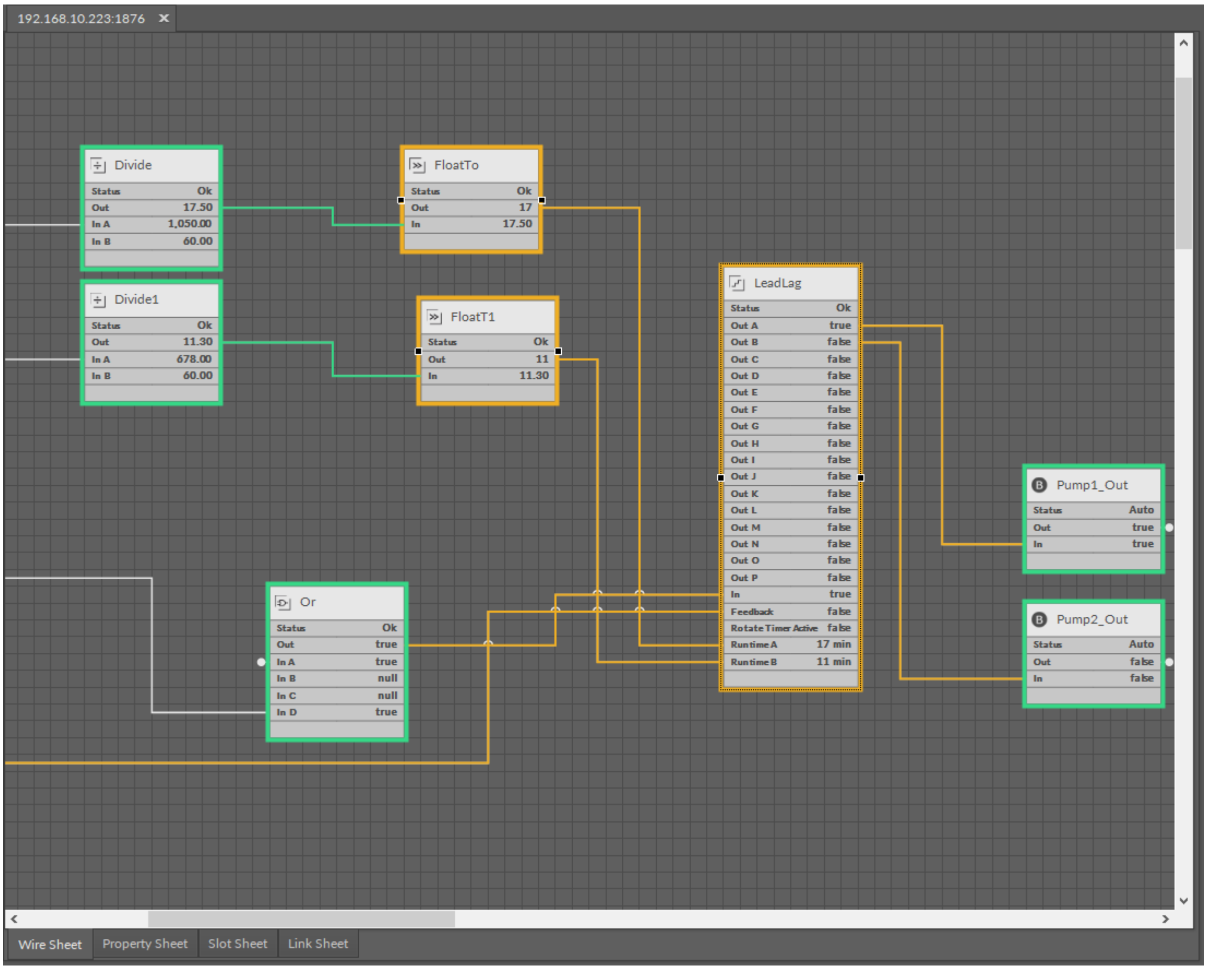


Figure 114. The crossing selection

The cross selection method enables easy removal of all input links of the component without a need to individually select each link. For this purpose drag a narrow rectangle across the component's links that are to be selected (see the figures below). All selected links (orange color) can be deleted with a Delete key or a context menu.

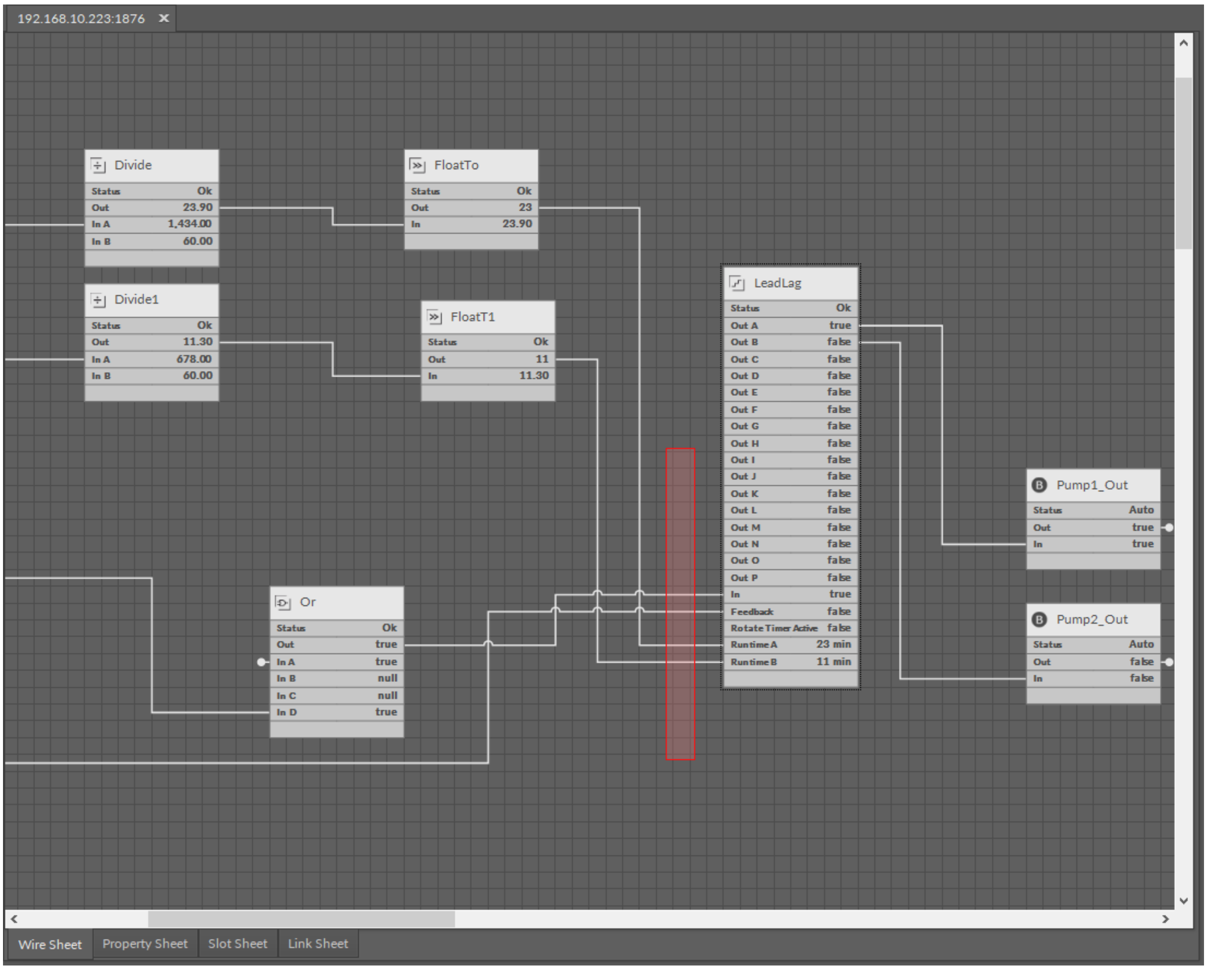
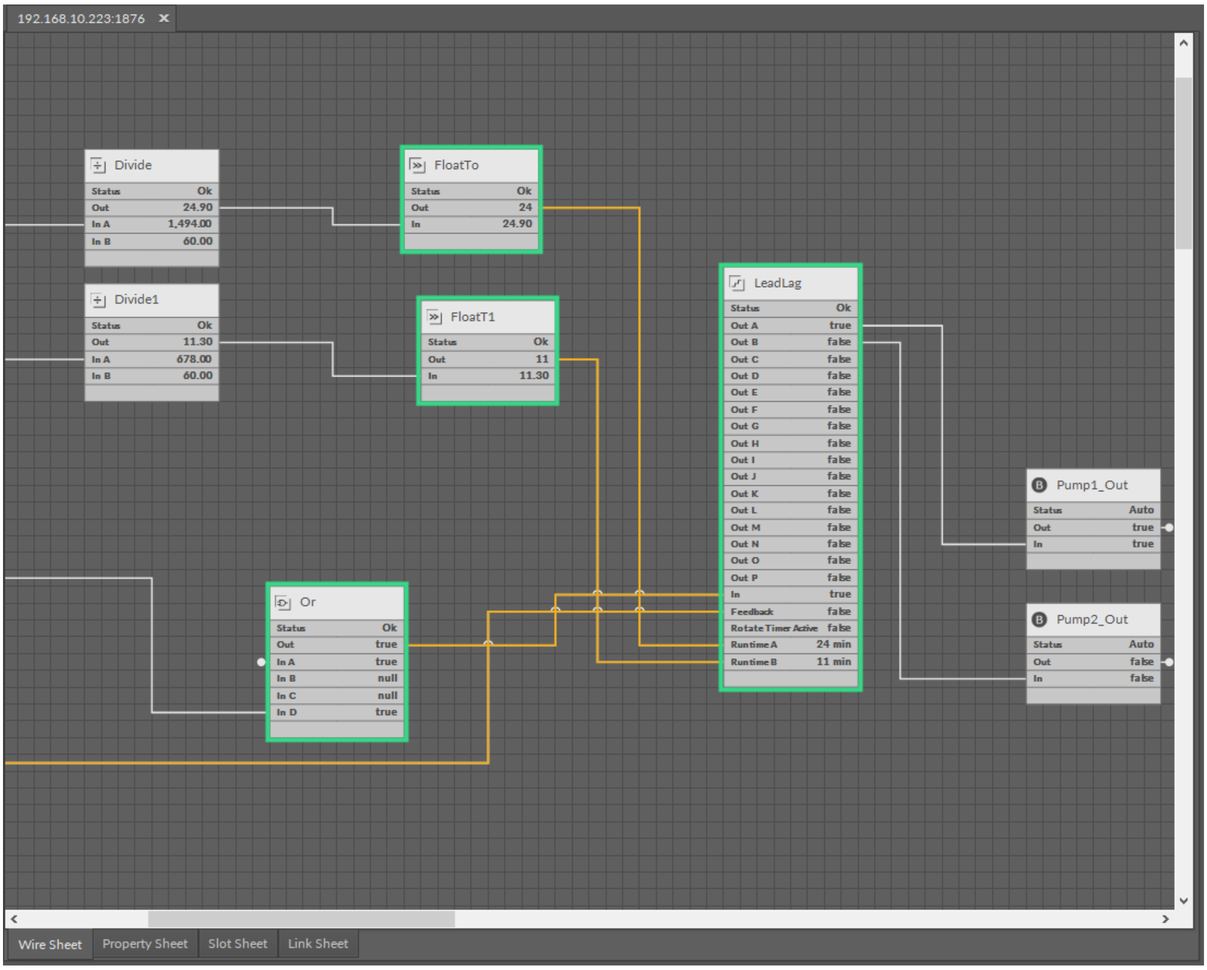


Figure 115. The links crossing selection



Divide

Status	Ok
Out	24.90
In A	1,494.00
In B	60.00

FloatTo

Status	Ok
Out	24
In	24.90

Divide1

Status	Ok
Out	11.30
In A	678.00
In B	60.00

FloatT1

Status	Ok
Out	11
In	11.30

LeadLag

Status	Ok
Out A	true
Out B	false
Out C	false
Out D	false
Out E	false
Out F	false
Out G	false
Out H	false
Out I	false
Out J	false
Out K	false
Out L	false
Out M	false
Out N	false
Out O	false
Out P	false
In	true
Feedback	false
Rotate Timer Active	false
Runtime A	24 min
Runtime B	11 min

Pump1_Out

Status	Auto
Out	true
In	true

Pump2_Out

Status	Auto
Out	false
In	false

Or

Status	Ok
Out	true
In A	true
In B	null
In C	null
In D	true

Figure 116. The links crossing selection

9.2.4 Adding and Removing Components in the Wire Sheet

Adding Components

There are three methods of adding a new component to the Wire Sheet view:

- dragging it from the Device Kits window;
- copying it in the Device Kits window and pasting it in the Wire Sheet tab;
- duplicating a component already added to the Wire Sheet (it is an option in the context menu).

If the component is dragged or pasted to the Wire Sheet, it is released where the mouse points. However, if the component has not been added directly to the Wire Sheet but, for example, placed in the Workspace Tree window, its position in the Wire Sheet will be the same as original component (including cascade offset) in case of pasting or duplicating. In case the component has been dragged from the Device Kits window to the Workspace Tree, its position in the Wire Sheet is (0, 0), it will appear in the upper left corner.

In all cases added components are slightly relocated relative to original components, so they do not overlap. The iSMA Tool uses a cascade mechanism here.

Removing Components

A component in the Wire Sheet view can be removed after selecting it and pressing the Delete key or choosing the Delete option from the context menu. After removing the component, the iSMA Tool will automatically remove all links connected to the removed component. If necessary, the remaining links will be automatically redrawn to use the space remaining after the removed component.

9.2.5 Adding and Removing Links in the Wire Sheet

Adding Links

According to Linking Components, a link is a way of data exchange between components. The link defines which components will exchange data. It precisely defines which parameters will be taken into account on both sides of the connection, and it determines the direction of data exchange, i.e., where the parameter will be read, and where it will be saved. In the Wire Sheet view the link is a broken line connecting slots, which exchange data.

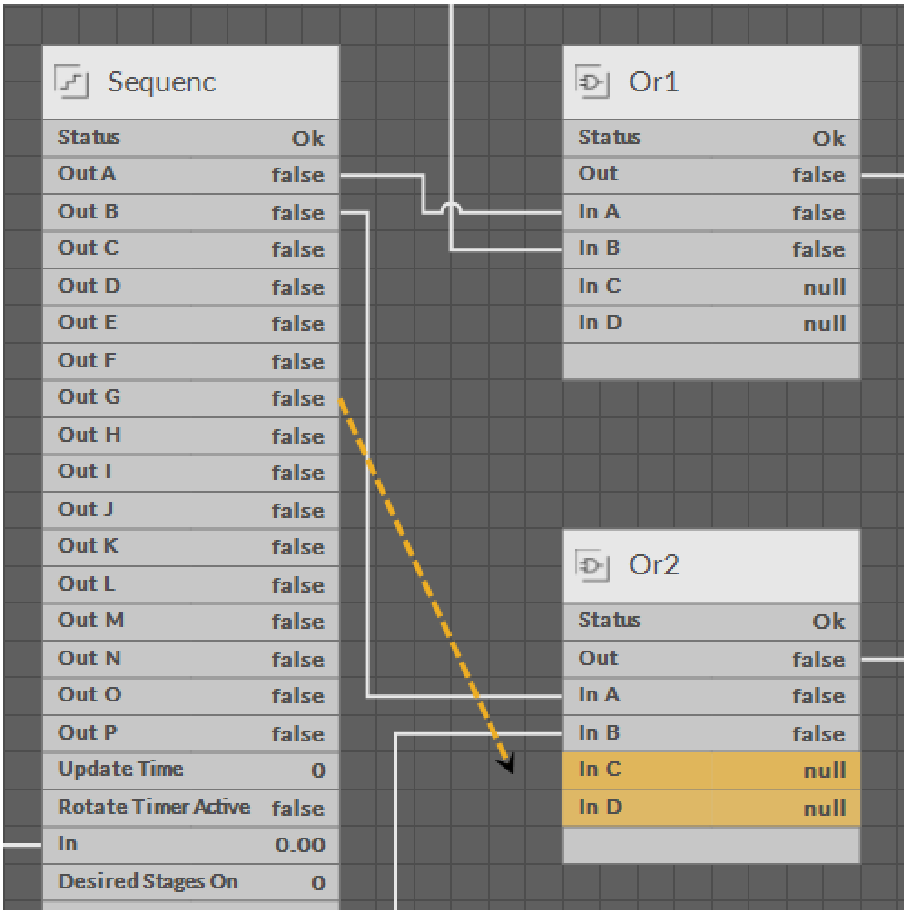
A component typically has at least three slots. They can be input slots, which can receive new data sent via the link, and output slots, presenting a value which is a result of calculations performed by the component. The output value can be sent further via the link. For the data exchange between slots through the link to be possible, a type of data sent and received must be the same. It has an impact on a link creation process, which can start at any slot, but the link connection finishing slot must be of the same type as the source slot. The iSMA Tool supervises this compatibility requirement, making it impossible to link slots of varying types.

The link can be added in the Wire Sheet using the Link Mark and Link From options (described in detail in Single Device Functions). The Link Mark option defines a beginning component for the link and its output slot. The Link From option defines an end component and its input slot.

In the Wire Sheet view links can also be created using a graphic method. In order to create a link graphically, click the selected output slot and holding the click drag the mouse to the selected input slot, then release. The iSMA Tool automatically colors the fitting input slots in orange. The intensity of coloring depends on the distance of the mouse pointer (see the figure below).

A dialog window shows the selected output slot (in the left column) and input slots available to select (in the right column). The available input slots are highlighted in the right column. Selecting the input slot has to be confirmed by pressing Ok.

Creating a link, regardless of a method used, is directly saved to the controller.



Sequenc

Status	Ok
Out A	false
Out B	false
Out C	false
Out D	false
Out E	false
Out F	false
Out G	false
Out H	false
Out I	false
Out J	false
Out K	false
Out L	false
Out M	false
Out N	false
Out O	false
Out P	false
Update Time	0
Rotate Timer Active	false
In	0.00
Desired Stages On	0

Or1

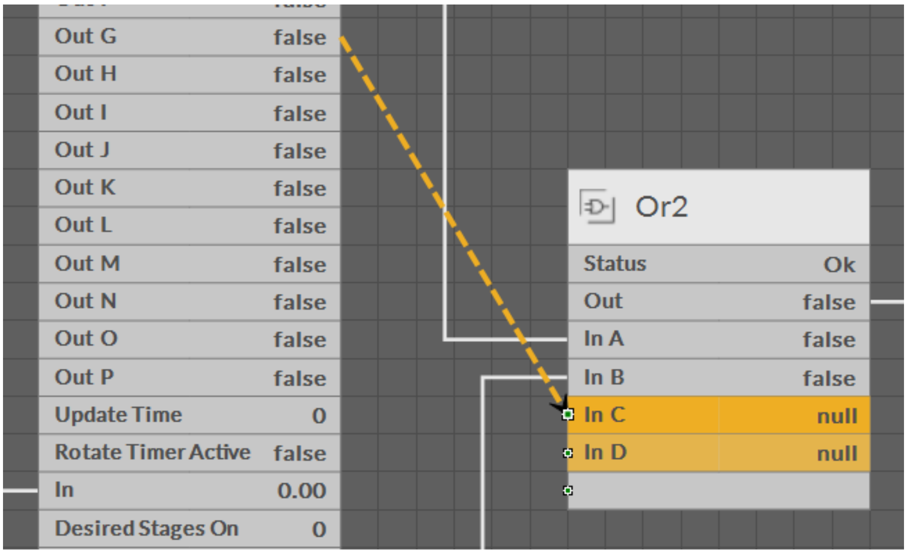
Status	Ok
Out	false
In A	false
In B	false
In C	null
In D	null

Or2

Status	Ok
Out	false
In A	false
In B	false
In C	null
In D	null

Figure 117. Linking

In the figure above, the link has been started in the Out G slot of the Sequenc component and can be finished in one of the Or2 component slots, marked by the iSMA Tool in orange (In C, In D). To finish the link, release the click over the selected output slot (the green hooks will be helpful, which will pull the mouse pointer to the chosen slot).



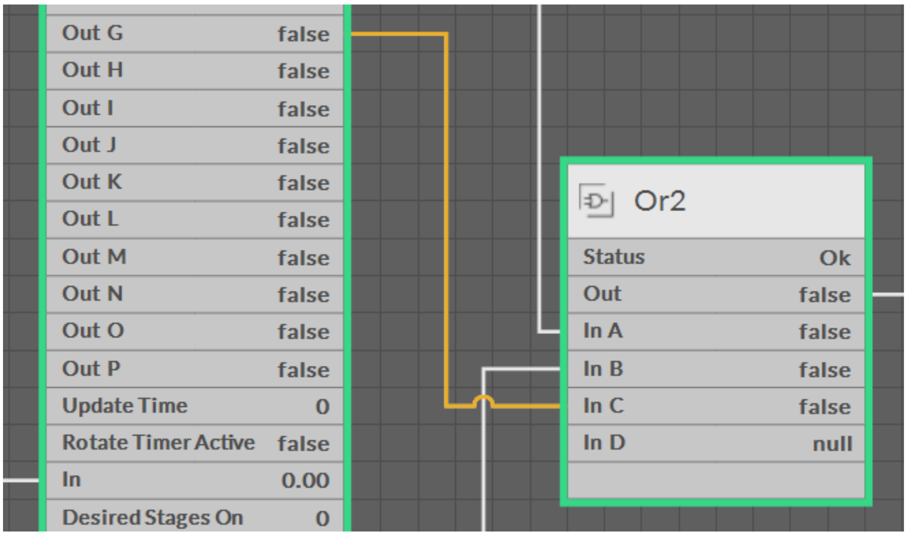
Out G	false
Out H	false
Out I	false
Out J	false
Out K	false
Out L	false
Out M	false
Out N	false
Out O	false
Out P	false
Update Time	0
Rotate Timer Active	false
In	0.00
Desired Stages On	0

Or2

Status	Ok
Out	false
In A	false
In B	false
In C	null
In D	null

Figure 118. Linking

The link drawing process described above can be carried out in the opposite way, i.e., first defining the In C slot of the Or2 component and drawing it in the direction of the Sequenc component. The end effect is identical (see the figure below).



Out G false
Out H false
Out I false
Out J false
Out K false
Out L false
Out M false
Out N false
Out O false
Out P false
Update Time 0
Rotate Timer Active false
In 0.00
Desired Stages On 0

Or2
Status Ok
Out false
In A false
In B false
In C false
In D null

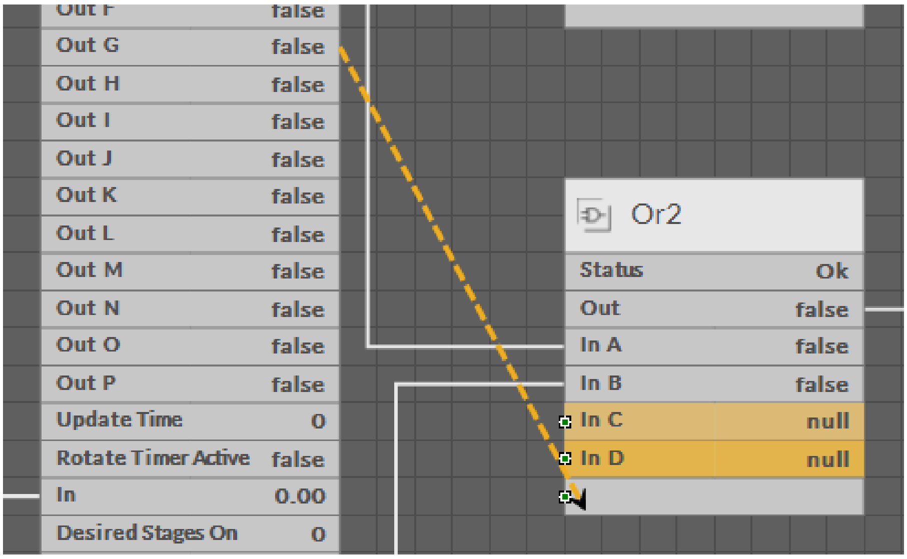
Figure 119. Linked components

The newly created link is automatically marked as selected, therefore it can be instantly deleted using the Delete key.

By default, not all slots of a particular component are always visible in the Wire Sheet view. The hidden slots are not displayed in the component default view. A hidden slot becomes visible once it is an end slot for the link. The link to hidden slot may be created using special linking window discussed in Chapter 11.5 Linking. The linking window can be initiated in two ways:

- using the Link Mark and Link From options from the context menu on a component;
- drawing a link between empty slots of components (last row).

Drawing the link to the empty slot, opens a dialog window, which shows the selected output slot (in the left column) and input slots available to select (in the right column).



Out F false
Out G false
Out H false
Out I false
Out J false
Out K false
Out L false
Out M false
Out N false
Out O false
Out P false
Update Time 0
Rotate Timer Active false
In 0.00
Desired Stages On 0

Or2
Status Ok
Out false
In A false
In B false
In C null
In D null

Figure 120. Drawing a link to a hidden slot

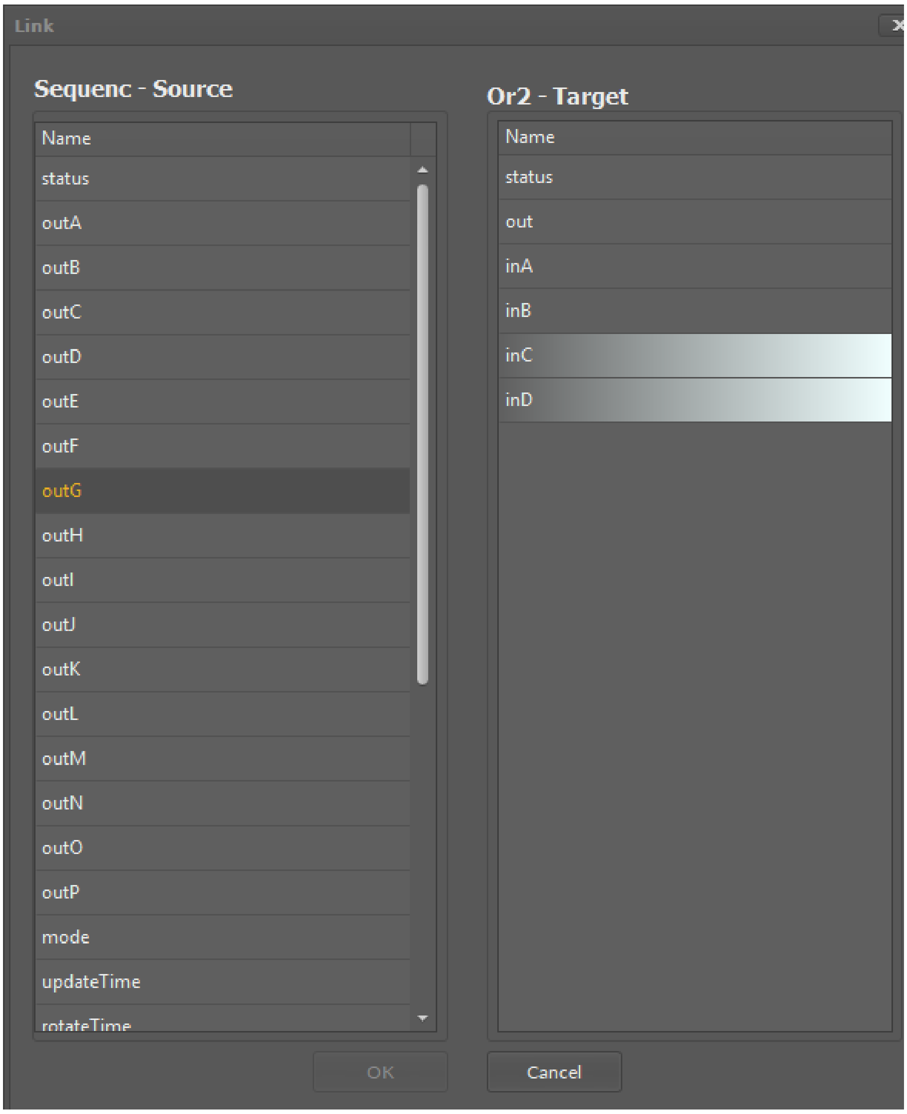


Figure 121. Linking dialog window

The available input slots are highlighted in the right column. Selecting the input slot has to be confirmed by pressing Ok.

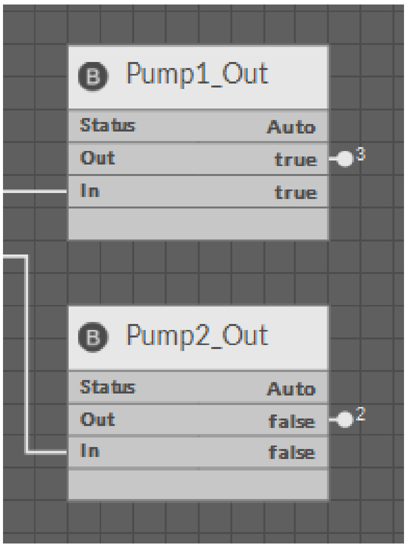
Creating a link, regardless of a method used, is directly saved to the controller.

Removing Links

A link in the Wire Sheet view can be removed after selecting it and pressing Delete key or choosing option Delete link(s) from the context menu. If necessary, the remaining links will be automatically redrawn to use the space remaining after the removed one.

Links Between Wire Sheet Tabs

Each Wire Sheet tab displays components belonging to a common parent component. More complex applications require to create a logical connection between components belonging to different parents, therefore displayed in different Wire Sheet tabs. For this reason links between components belonging to different parents are displayed as full circles coming out of linked slots. It means that the further part of such link is continued in another Wire Sheet tab (see the figure below).



Pump1_Out

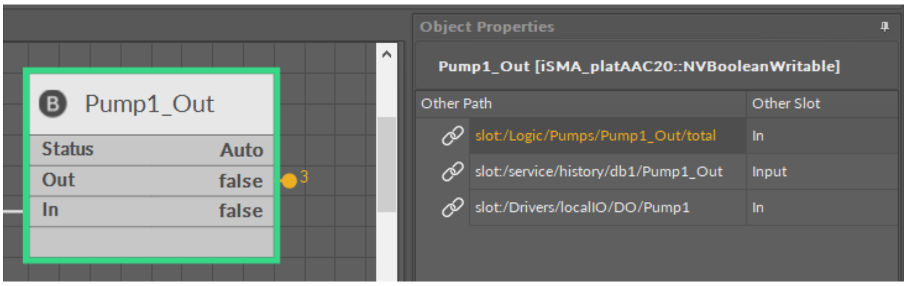
Status	Auto
Out	true
In	true

Pump2_Out

Status	Auto
Out	false
In	false

Figure 122. Two components with external links

Additionally, in case there is more than one link coming out from one output slot, a number of links is displayed next to the link circle. In the figure above there are three links coming out to the other Wire Sheet from the Pump1_Out component's Out slot. Detailed information about the links' directions, slots, and components can be obtained in the Object Properties window after clicking the links circle (see the figure below).



Pump1_Out

Status	Auto
Out	false
In	false

Object Properties

Pump1_Out [iSMA_platAAC20::NVBooleanWritable]

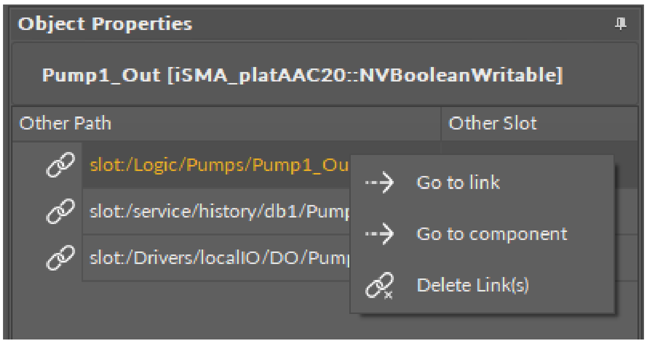
Other Path	Other Slot
slot:/Logic/Pumps/Pump1_Out/total	In
slot:/service/history/db1/Pump1_Out	Input
slot:/Drivers/localIO/DO/Pump1	In

Figure 123. Component with external link

The Object Properties window (on the right) shows details of three links with full location paths of target components and slot names.

In the Object Properties window it is possible to choose the following options from the context menu for each link:

- Go to link: go to the Wire Sheet tab of the parent of the target component;
- Go to component: go to the Property Sheet tab of the target component;
- Delete link: remove a chosen link.



Object Properties

Pump1_Out [iSMA_platAAC20::NVBooleanWritable]

Other Path	Other Slot
slot:/Logic/Pumps/Pump1_Ou	
slot:/service/history/db1/Pump	
slot:/Drivers/localIO/DO/Pum	

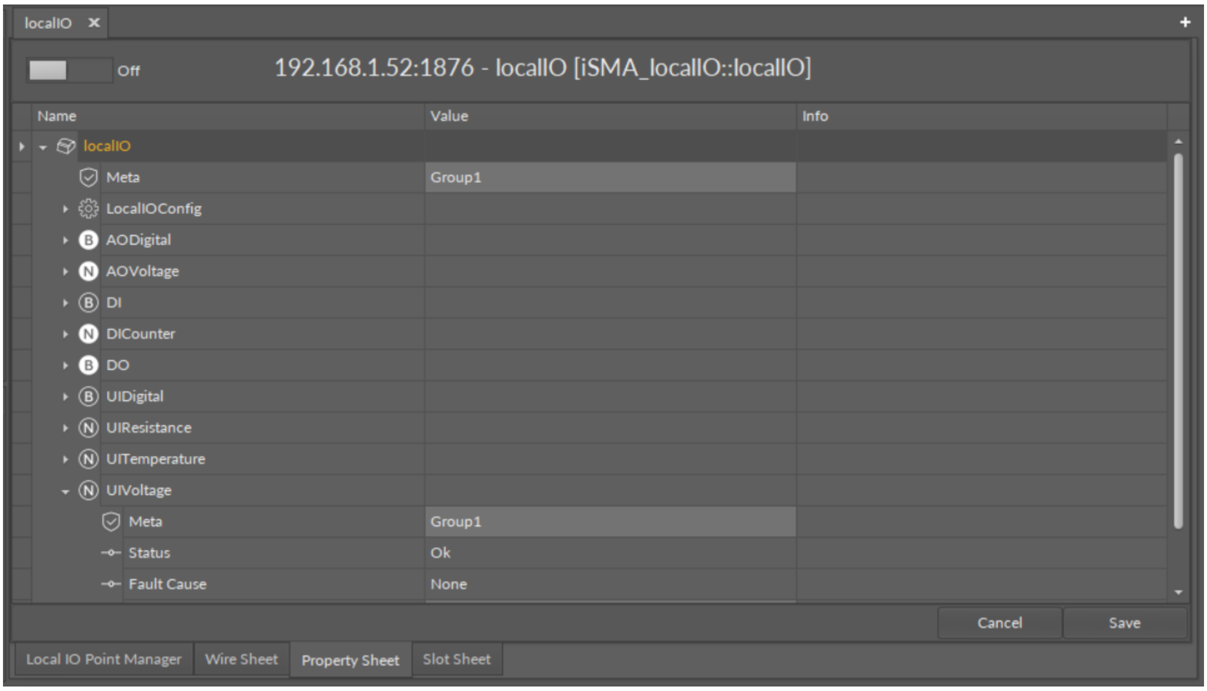
Figure 124. Link options in the Object Properties window

9.3 Property Sheet

A Property Sheet is a tabular view listing all of the selected component's slots, its inferior components, and their slots. The Property Sheet provides information about slots' names, their current values, units, and ranges .

The Property Sheet view allows to select only one component at a time to perform basic functions such as copying, duplicating, and removing.

Each component in the Property Sheet view is displayed in an expandable tree structure. Slots, which values can be edited, are highlighted with a lighter background.



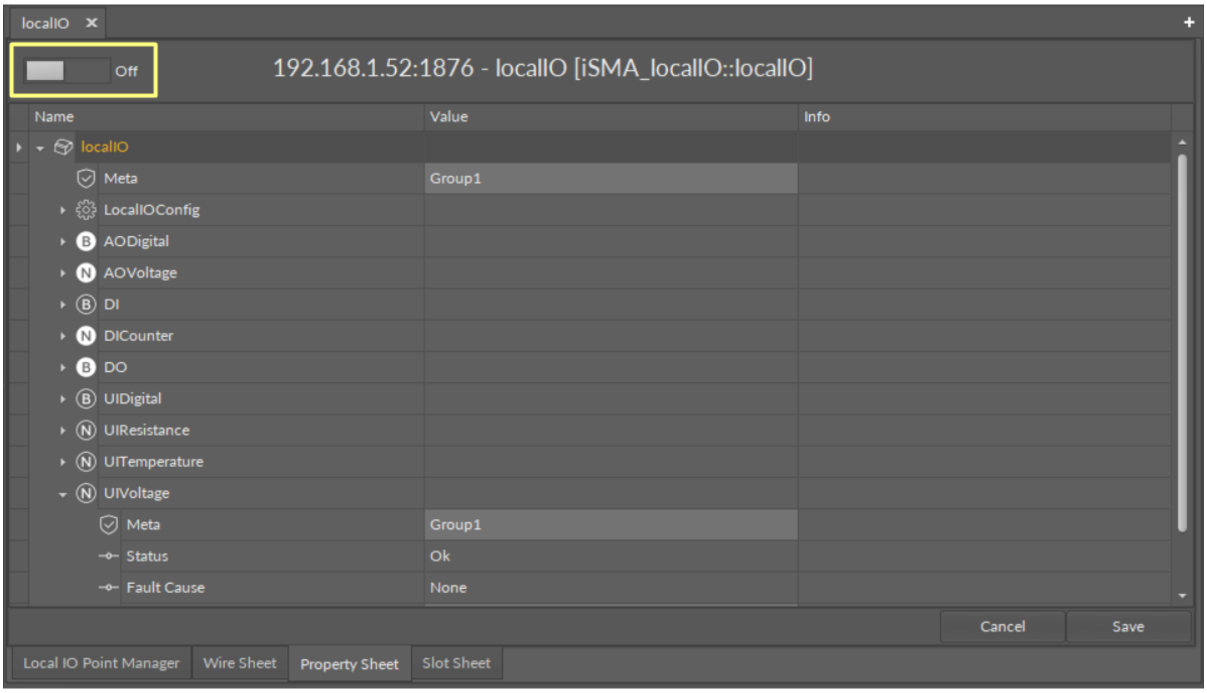
192.168.1.52:1876 - localIO [iSMA_localIO::localIO]

Name	Value	Info
localIO		
Meta	Group1	
LocalIOConfig		
AODigital		
AOVoltage		
DI		
DICounter		
DO		
UIDigital		
UIResistance		
UITemperature		
UIVoltage		
Meta	Group1	
Status	Ok	
Fault Cause	None	

Figure 125. The Property Sheet view

9.3.1 Saving in Property Sheet View

There are two modes of saving edited values–manual and automatic. Modes are switched with a binary key, located in the upper left corner of the Property Sheet view.



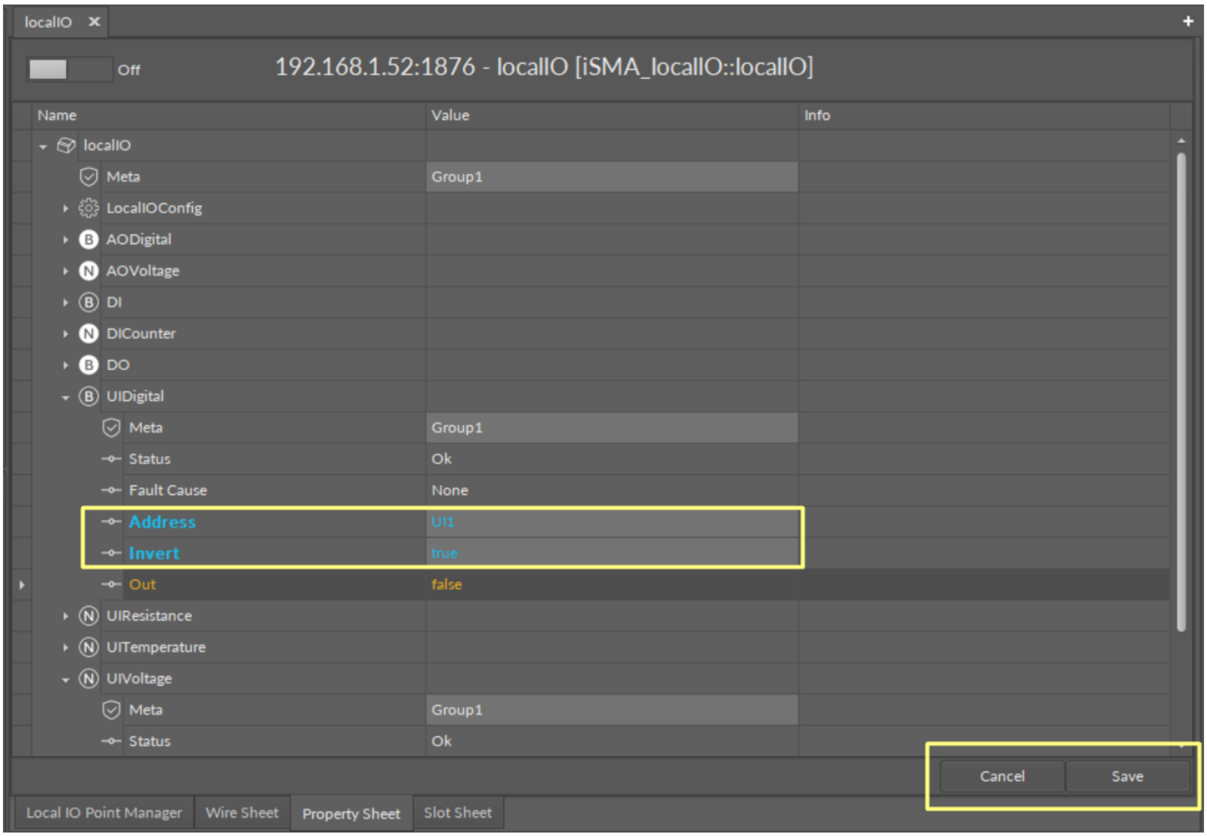
192.168.1.52:1876 - localIO [iSMA_localIO::localIO]

Name	Value	Info
localIO		
Meta	Group1	
LocalIOConfig		
AODigital		
AOVoltage		
DI		
DICounter		
DO		
UIDigital		
UIResistance		
UITemperature		
UIVoltage		
Meta	Group1	
Status	Ok	
Fault Cause	None	

Figure 126. Autosave key

If the key is switched to On, the automatic saving mode is enabled. Each edited value is directly saved to the controller.

If the key is switched to Off, the manual saving mode is enabled. Each edited value has its font color changed to blue and it needs to be saved manually with a Save key displayed at the bottom of the Property Sheet tab. Alternatively, the edited value may be saved by confirming it with the Enter key. Only after the edited values are saved manually, they are sent to the controller.

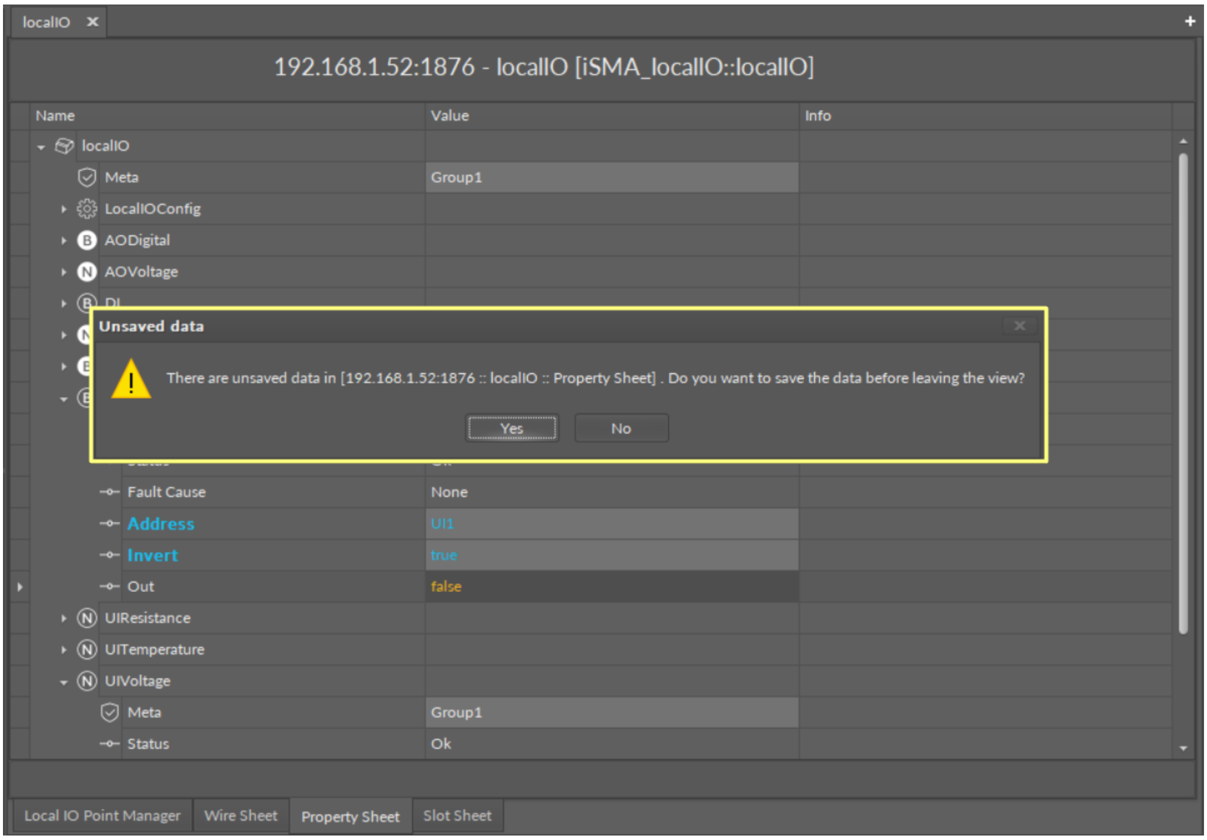


192.168.1.52:1876 - localIO [iSMA_localIO::localIO]

Name	Value	Info
localIO		
Meta	Group1	
LocalIOConfig		
AODigital		
AOVoltage		
DI		
DICounter		
DO		
UIDigital		
Meta	Group1	
Status	Ok	
Fault Cause	None	
Address	UI1	
Invert	true	
Out	false	
UIResistance		
UITemperature		
UIVoltage		
Meta	Group1	
Status	Ok	

Figure 127. Manual saving mode

The iSMA Tool warns the user of unsaved values in the manual saving mode if the user is trying to close the Property Sheet tab, disconnect a device, or close the program (see the figure below).



Name	Value	Info
localIO		
Meta	Group1	
LocalIOConfig		
AODigital		
AOVoltage		
DI		
Fault Cause	None	
Address	UI1	
Invert	true	
Out	false	
UIResistance		
UITemperature		
UIVoltage		
Meta	Group1	
Status	Ok	

Unsaved data

There are unsaved data in [192.168.1.52:1876 :: localIO :: Property Sheet] . Do you want to save the data before leaving the view?

Figure 128. The manual saving alert

9.3.2 Adding and Removing Components in the Property Sheet

Adding Components

In order to add a new component to the Property Sheet view, drag it from the Device Kits window straight to the Property Sheet tab. As the Property Sheet tab can only be opened for a selected component (displaying a plain Property Sheet tab is not possible), the dragged component will always be added as a child of the selected component. If the dragged component is dropped in an unoccupied field in the Property Sheet view, it will be added to the main component of this view.

Removing Components

A component selected in the Property Sheet tab can be deleted with a Delete key or a Delete option in the context menu.

9.3.3 Creating Links

The only way of creating a link in the Property Sheet view is using the context menu with the Link Mark and Link From options discussed in the Linking section.

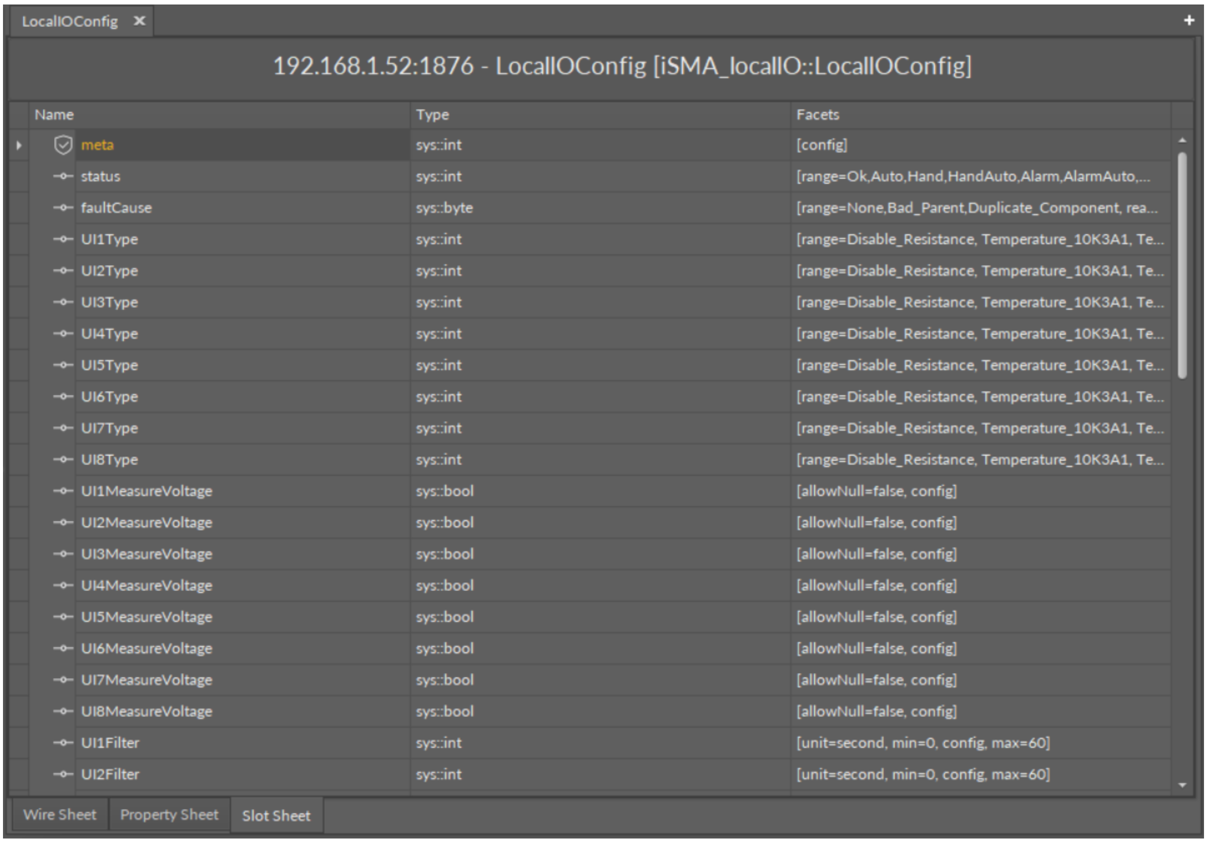
There is no possibility of removing a link in the Property Sheet view. To remove a link use the Wire Sheet view.

9.4 Slot Sheet

A Slot Sheet view displays a list of slots of a selected component. It is not a dynamic view– the values of component's slots are read once, when the Slot Sheet is opened, and there is no possibility to edit them.

The Slot Sheet view provides information about the slots of the selected component such as slot type, slot variable type, and slot facet interface.

The slot list contains all component's slots, including hidden slots which are not visible in the Wire Sheet view. For the LeadLag component the number of slots presented in the Slot Sheet view is much higher than visible in the Wire Sheet view (see the figure below).



192.168.1.52:1876 - LocalIOConfig [iSMA_localIO::LocalIOConfig]

Name	Type	Facets
meta	sys::int	[config]
status	sys::int	[range=Ok,Auto,Hand,HandAuto,Alarm,AlarmAuto,...
faultCause	sys::byte	[range=None,Bad_Parent,Duplicate_Component, rea...
UI1Type	sys::int	[range=Disable_Resistance, Temperature_10K3A1, Te...
UI2Type	sys::int	[range=Disable_Resistance, Temperature_10K3A1, Te...
UI3Type	sys::int	[range=Disable_Resistance, Temperature_10K3A1, Te...
UI4Type	sys::int	[range=Disable_Resistance, Temperature_10K3A1, Te...
UI5Type	sys::int	[range=Disable_Resistance, Temperature_10K3A1, Te...
UI6Type	sys::int	[range=Disable_Resistance, Temperature_10K3A1, Te...
UI7Type	sys::int	[range=Disable_Resistance, Temperature_10K3A1, Te...
UI8Type	sys::int	[range=Disable_Resistance, Temperature_10K3A1, Te...
UI1MeasureVoltage	sys::bool	[allowNull=false, config]
UI2MeasureVoltage	sys::bool	[allowNull=false, config]
UI3MeasureVoltage	sys::bool	[allowNull=false, config]
UI4MeasureVoltage	sys::bool	[allowNull=false, config]
UI5MeasureVoltage	sys::bool	[allowNull=false, config]
UI6MeasureVoltage	sys::bool	[allowNull=false, config]
UI7MeasureVoltage	sys::bool	[allowNull=false, config]
UI8MeasureVoltage	sys::bool	[allowNull=false, config]
UI1Filter	sys::int	[unit=second, min=0, config, max=60]
UI2Filter	sys::int	[unit=second, min=0, config, max=60]

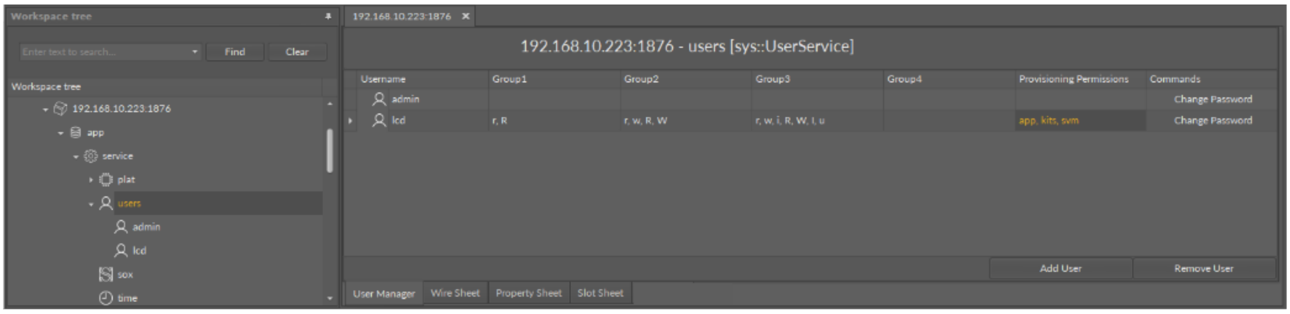
Figure 129. The Slot Sheet view

10 Services

This section outlines services available for Sedona-based devices.

10.1 User Manager

The iSMA Tool allows to manage users of the controller with a dedicated User Manager view. It allows to define a range of changes in controller application for each user. The User Manager is opened by double-clicking the user service in the Workspace Tree window.



192.168.10.223:1876 - users [sys::UserService]

Username	Group1	Group2	Group3	Group4	Provisioning Permissions	Commands
admin						Change Password
lcd	r, R	r, w, R, W	r, w, i, R, W, I, u		app, kits, svm	Change Password

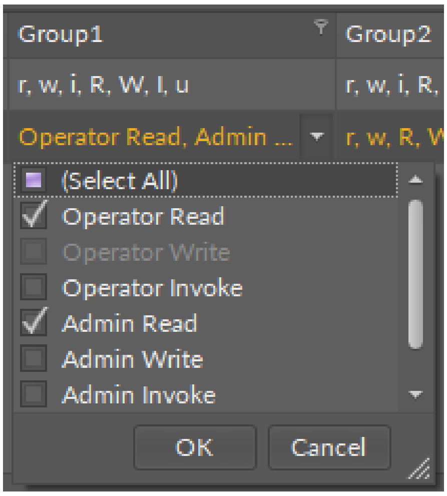
Figure 130. The User Manager view

The User Manager view shows the defined users, allows to edit their permissions and passwords, and makes it possible to add a new user and remove an already existing one.

The figure above shows two users defined in the controller, admin and lcd,, with different access permissions defined in table columns. The admin user has no restrictions (empty columns) and has full access to the controller, while the lcd user has defined restrictions in accordance with rules set by Sedona environment and represented by columns:

- Group1-Group4 shows permissions given to components assigned to these groups;
- Provisioning Permissions defines what services a particular user is able to use;
- Commands column has a button allowing to set a new password for a user.

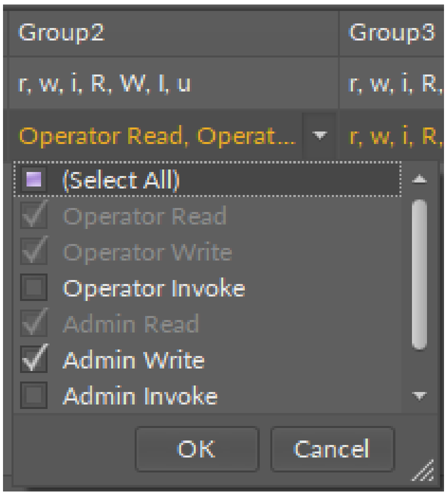
Double-clicking on each column opens a pop-up window with check boxes to edit users options and permissions.



Group1	Group2
r, w, i, R, W, I, u	r, w, i, R,
Operator Read, Admin ...	r, w, R, W

■ (Select All)
☑ Operator Read
□ Operator Write
□ Operator Invoke
☑ Admin Read
□ Admin Write
□ Admin Invoke

Figure 131. Users editing options



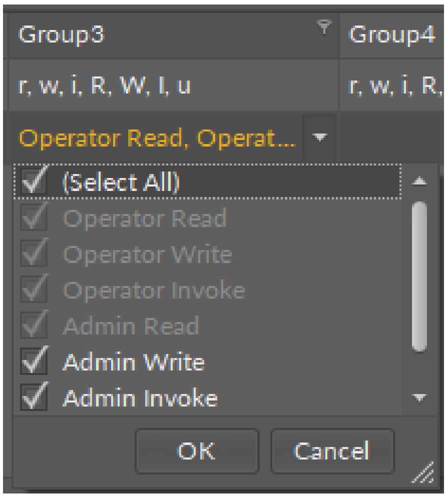
Group2 | Group3

r, w, i, R, W, I, u | r, w, i, R,

Operator Read, Operat... | r, w, i, R,

- ■ (Select All)
- ☑ Operator Read
- ☑ Operator Write
- ☐ Operator Invoke
- ☑ Admin Read
- ☑ Admin Write
- ☐ Admin Invoke

Figure 132. Users editing options



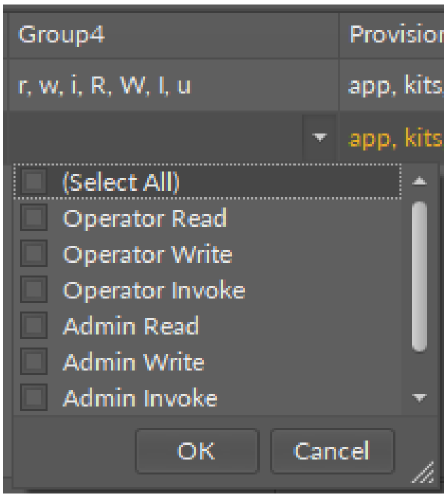
Group3 | Group4

r, w, i, R, W, I, u | r, w, i, R,

Operator Read, Operat...

- ☑ (Select All)
- ☑ Operator Read
- ☑ Operator Write
- ☑ Operator Invoke
- ☑ Admin Read
- ☑ Admin Write
- ☑ Admin Invoke

Figure 133. Users editing options



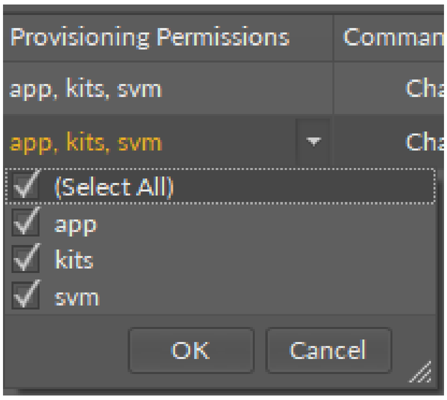
Group4 | Provisior

r, w, i, R, W, I, u | app, kits

app, kits

- ☐ (Select All)
- ☐ Operator Read
- ☐ Operator Write
- ☐ Operator Invoke
- ☐ Admin Read
- ☐ Admin Write
- ☐ Admin Invoke

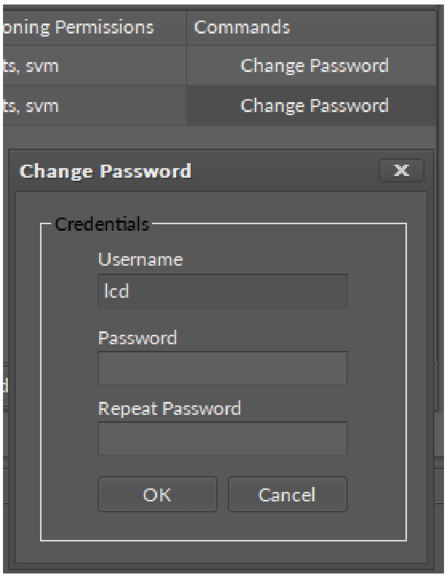
Figure 134. Users editing options



Provisioning Permissions	Comman
app, kits, svm	Cha
app, kits, svm	Cha

☑ (Select All)
☑ app
☑ kits
☑ svm

Figure 135. Users editing options



oning Permissions	Commands
ts, svm	Change Password
ts, svm	Change Password

Username: lcd

Figure 136. Users editing options

At the bottom of the User Manager tab are two buttons to add and remove users. In order to add a user, enter a username and a password. Adding a new user in the iSMA Tool will automatically create a component named the same as the username and expand the user list in the User Manager view.

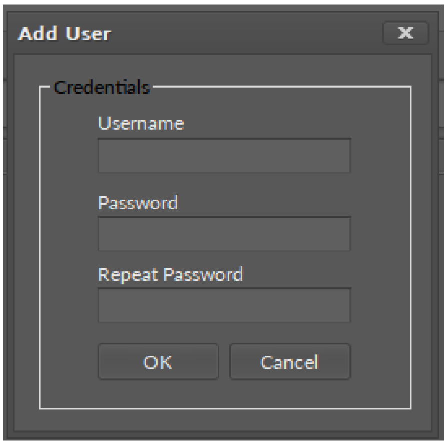


Figure 137. Dialog window for adding a user

Note: It is not recommended to change the username (component's name) after setting a password. If it is necessary to change the username, a new password should be set for it.

Note: In case a component representing a user is duplicated to create a new user with the same permissions, new user should also have a new password set.

Removing of a particular user may be done using two methods: either by removing the row in User Manager view, selecting it and pressing the Remove user button, or by removing the component representing that user.

10.2 Historical Data

The iSMA Tool allows to perform an analysis of historical trends by using two views, History Chart and History Table. The first view allows to work with a single trend or many trends on a graph versus time. The second view is a table of registered samples of one of the processes.

Accessing both views is possible from the Workspace Tree at app -> service -> history -> db1. Under the db1 component, which is a historical database, there are historical variables registered in the iSMA Tool shown by components (see the figure below).

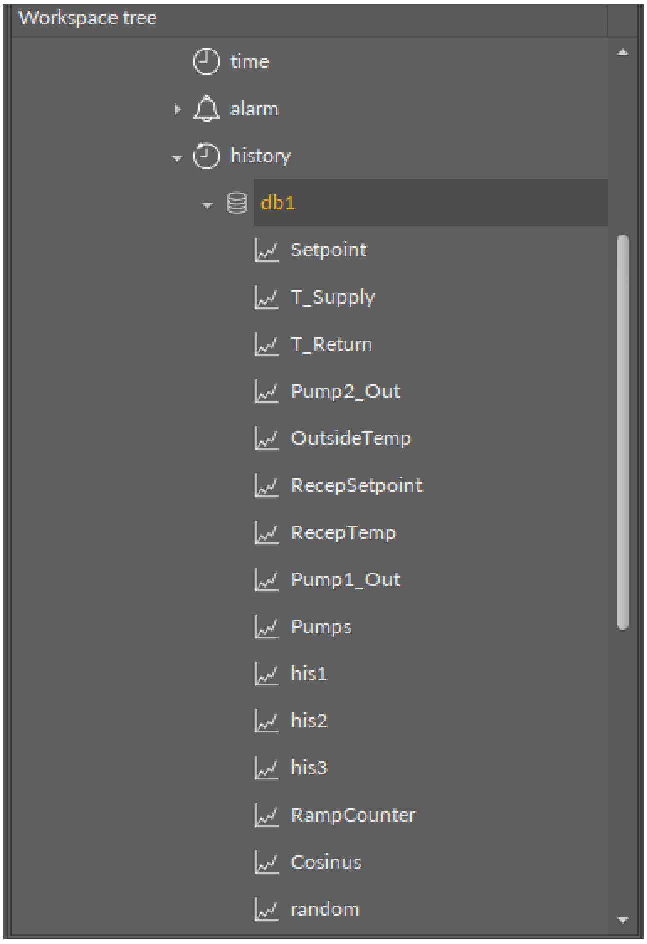


Figure 138. The db1 component's tree

Opening any component under the db1 component will open the selected component in the main screen along with the History Chart and History Table.

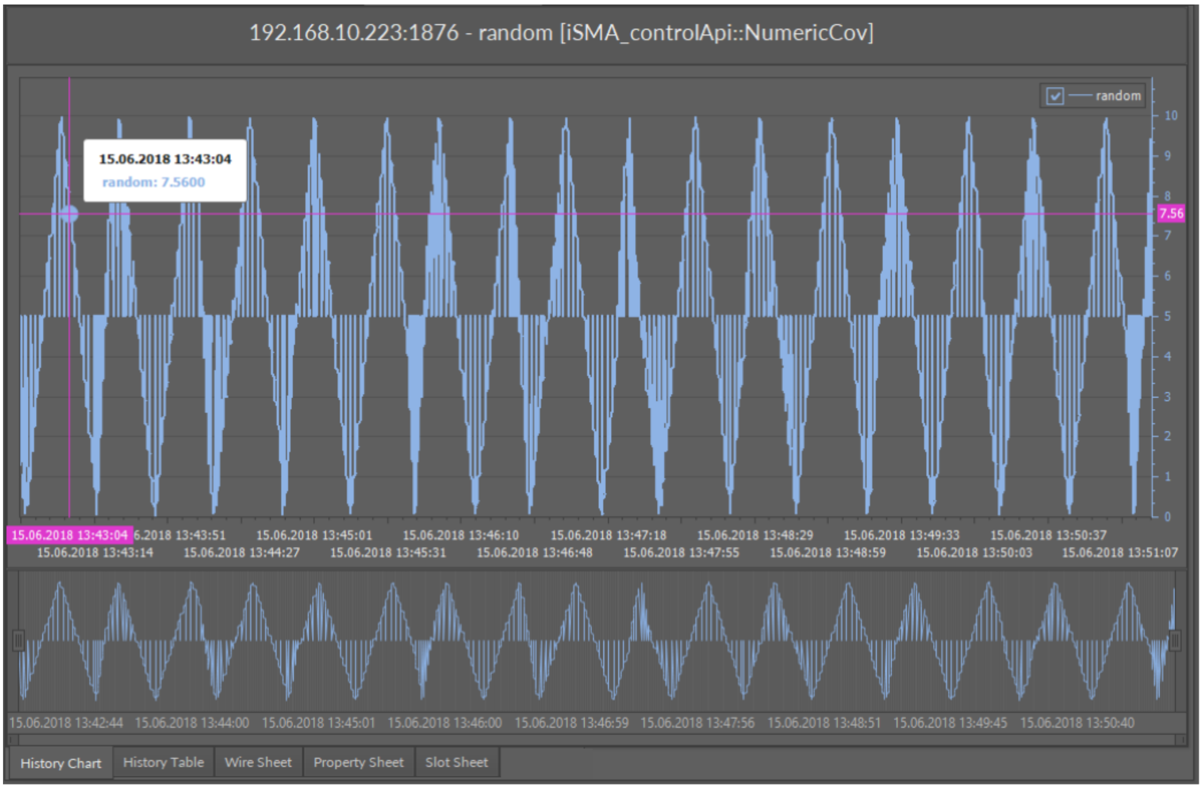


Figure 139. The History Chart view

The second method of working with historical data is expanding the db1 component and dragging the component we are interested in from the Workspace Tree to an empty History Chart view. The figure below shows the empty History Chart view of the db1 component before selecting the component to present its historical data.

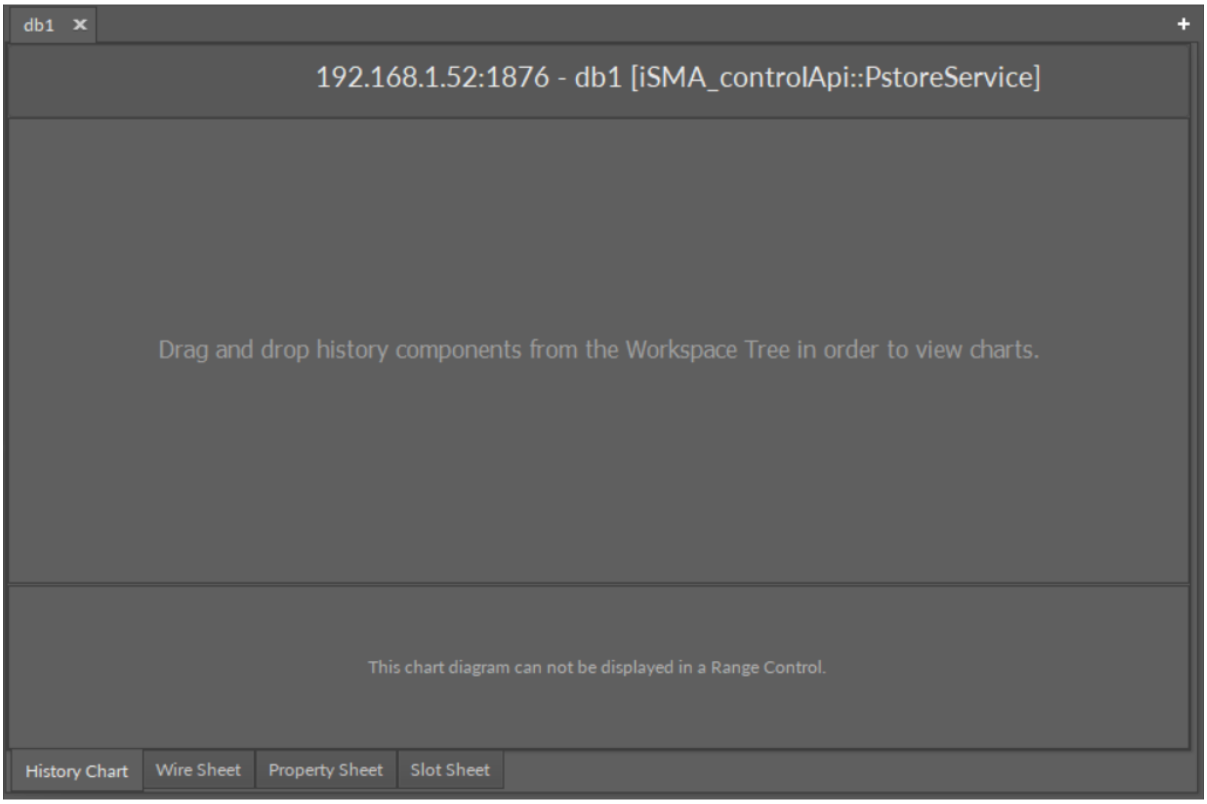

db1
192.168.1.52:1876 - db1 [iSMA_controlApi::PstoreService]
Drag and drop history components from the Workspace Tree in order to view charts.
This chart diagram can not be displayed in a Range Control.
History Chart
Wire Sheet
Property Sheet
Slot Sheet

Figure 140. An empty History Chart

10.2.1 Exporting and Printing Program (Preview)

The Preview program is launched with the Export/Print option in the graph's or history table context menu. Exporting or printing the graph or table requires defining few of the parameters described below.

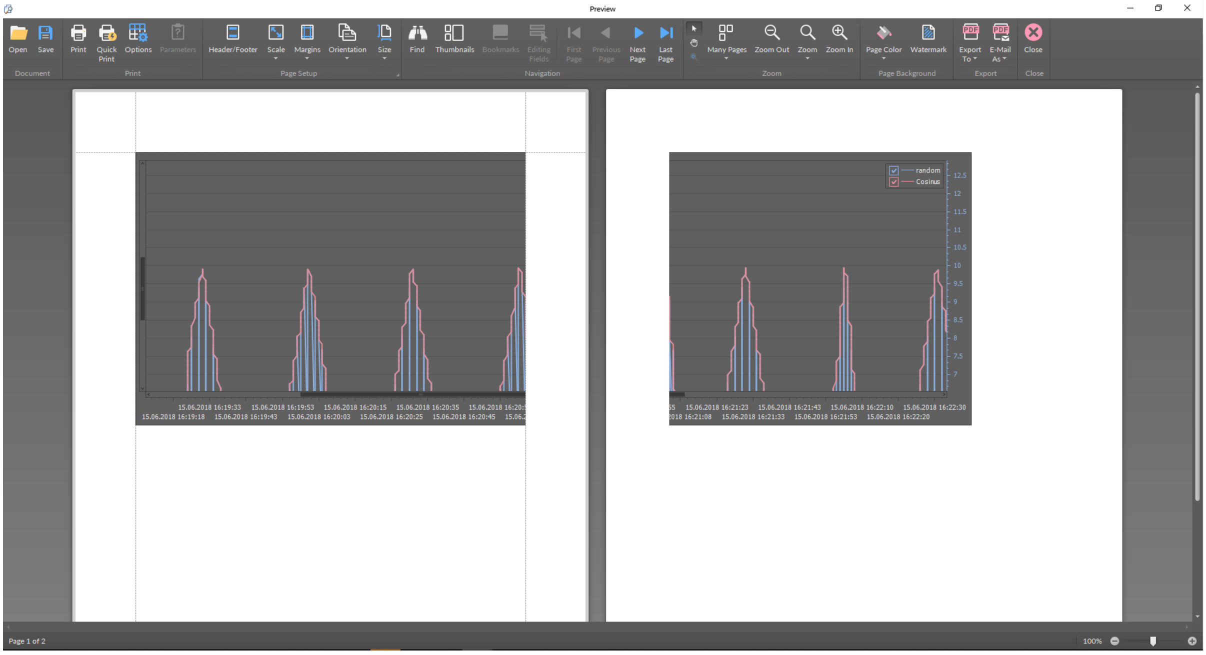
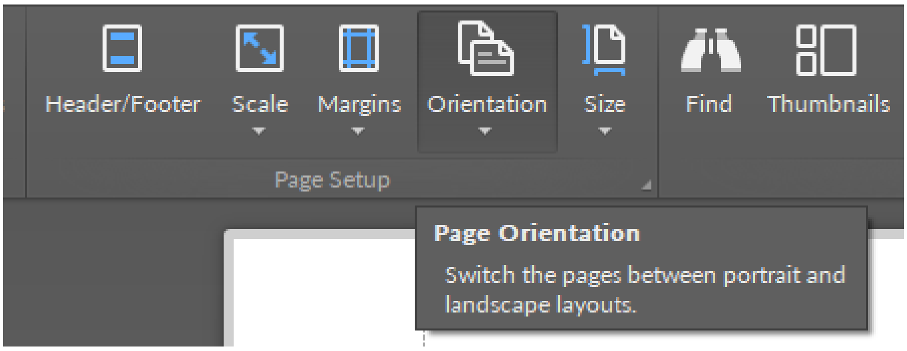


Figure 141. The Preview program

The program toolbar offers functions to adjust the graph to be exported or printed. The basic editing options are the following:

- Header/Footer: allows to define a header or footer for the document;
- Scale: allows to scale the contents to a page size;
- Orientation: changes page orientation to horizontal or vertical;



Page Orientation

Switch the pages between portrait and landscape layouts.

Figure 142. The Orientation button on a toolbar

- Size: adjusts margins and the page size
- Page Color: allows to set the background page color;
- Watermark: allows to add the watermark to the page background.

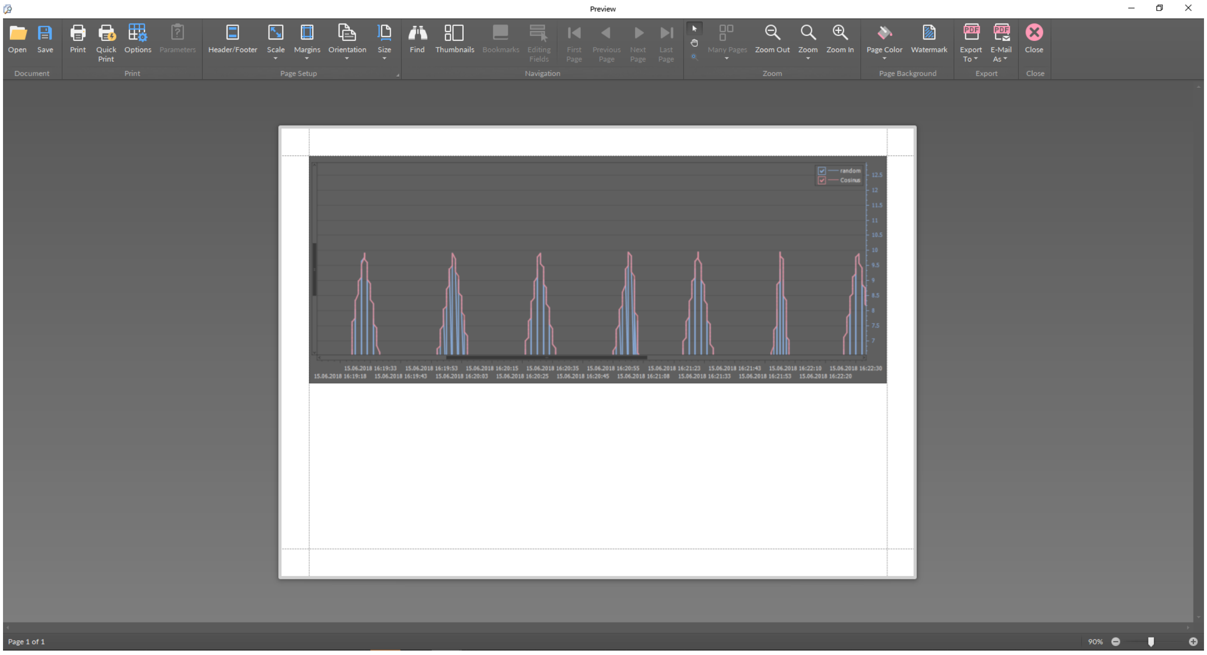


Figure 143. The graph adjusted to exporting/printing

Once the document is edited and adjusted to be printed or exported, the toolbar includes further actions buttons:

- Open: allows to open a previously saved file to be exported or printed;
- Save: allows to save a file for further action in available formats: .pdf, .html, .docx, .xlsx, .bmp, .gif, .jpeg, .png, .tiff, .emf, and .wmf;
- Print: launches printing and allows to choose the printer and its parameters;
- Quick Print: launches printing with default printing settings;
- Export To: launches exporting the file to available formats: .pdf, .html, .docx, .xlsx, .bmp, .gif, .jpeg, .png, .tiff, .emf, and .wmf;
- Email As: exports the file and attaches it to an e-mail message.

10.2.2 History Chart View

The History Chart view is a view dedicated for a graphic presentation of historical data versus time, commonly called a graph. A graph is a line connecting subsequent samples, defined in a time interval and a range of values. The graph is located in the upper and main area of the view. Below there is a smaller area with a preview of the whole available range saved in the controller.

In the preview area there is a possibility of setting time limits against which the trend will be drawn in the main area above.

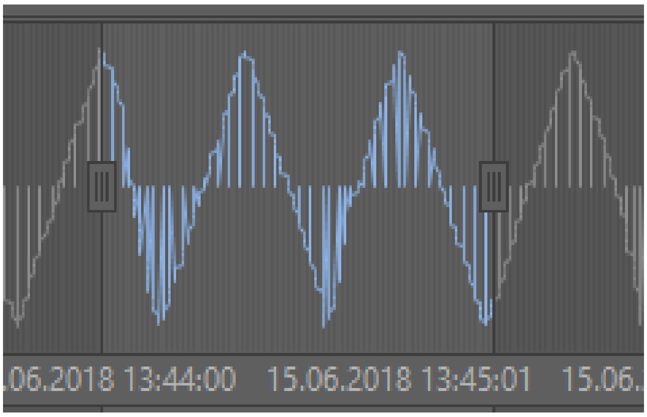


Figure 144. Setting time limits

After setting the time limits, the corresponding graph has been significantly narrowed and looks as shown on the figure below.

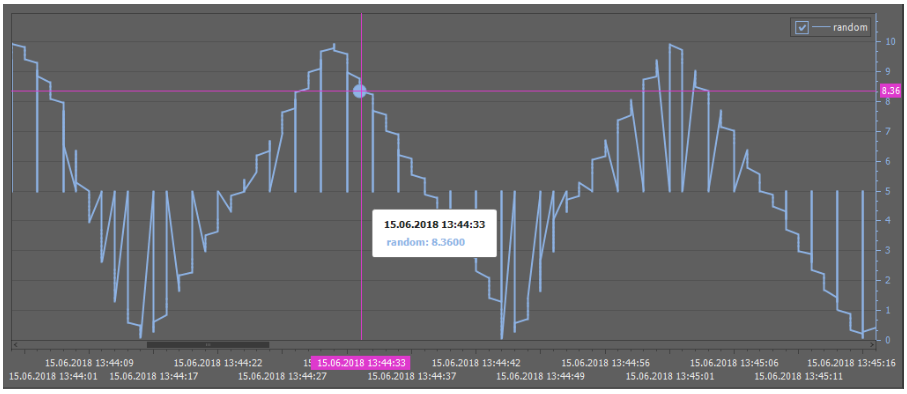


Figure 145. A graph after setting time limits

To analyze values on the History Chart move the mouse pointer over the registered samples–it shows their precise values. Additionally, the auxiliary lines show values on vertical and horizontal axes of the graph.

Clicking and holding the click allows to move the graph, while using the mouse wheel allows to zoom in/out the fragment of a graph. Each change of the graph's range is updated in the preview area under the graph.

It is possible to add another trend to the observed chart and display them both simultaneously for analysis. To add another trend it needs to be dragged from the Workspace Tree, from under the db1 tree to the opened chart. The figure below shows a situation after adding the RampCounter trend.

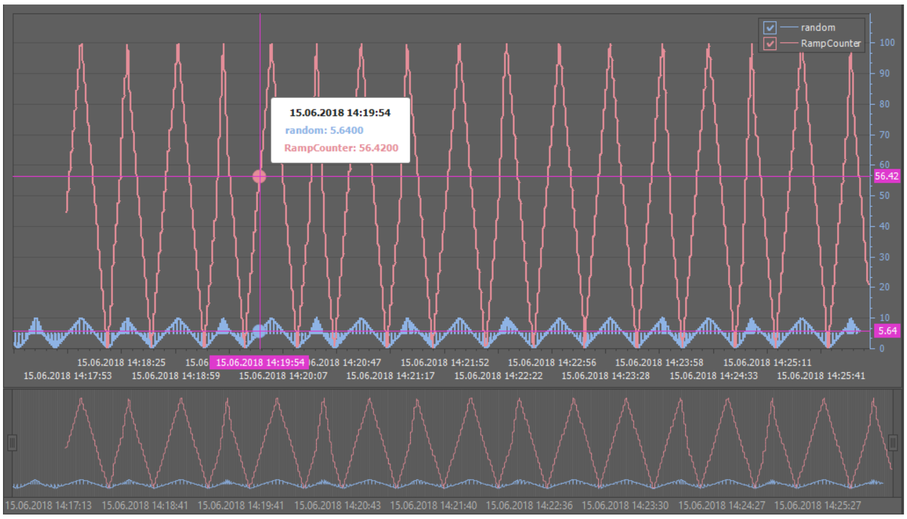


Figure 146. Two trends displayed in the History Chart

After adding another trend, the upper graph and the preview area have been updated accordingly. As the two trends are displayed simultaneously, the auxiliary lines track and show registered samples for the two trends independently.

By default, after adding a second trend the value scale is common for both processes, but individual scales can be displayed for each of historical processes. This is allowed by the Enable/Disable Secondary Axes option in the context menu of the graph.

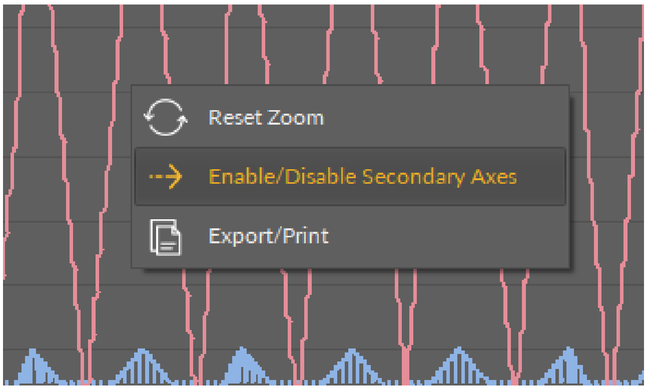


Figure 147. A context menu of a graph

After initiating an individual oy axis, for each of the processes, trends are automatically rescaled to cover as big part of the graph area as possible (see the figure below).

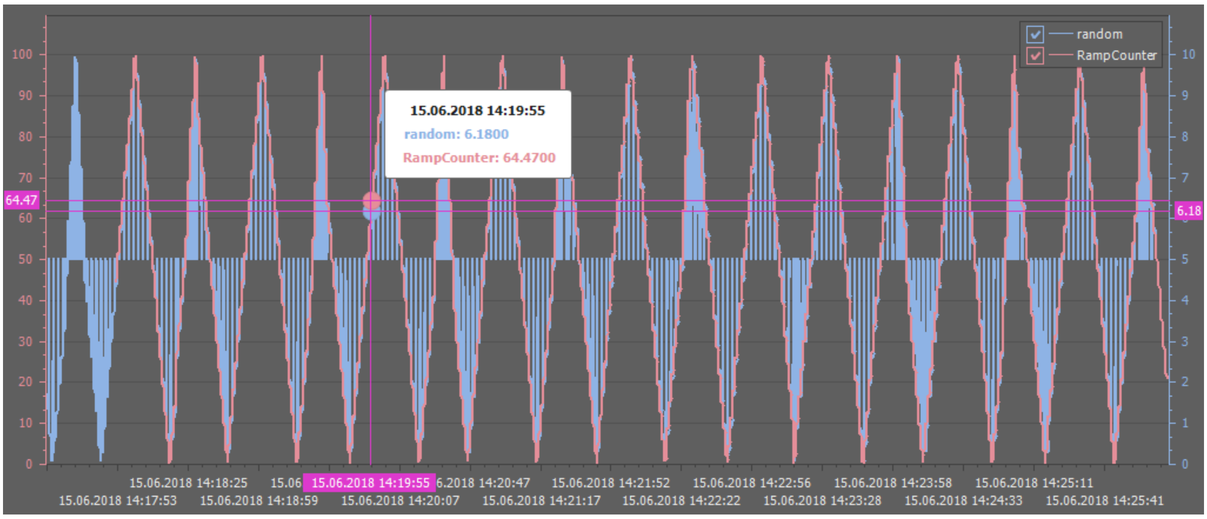


Figure 148. A rescaled graph

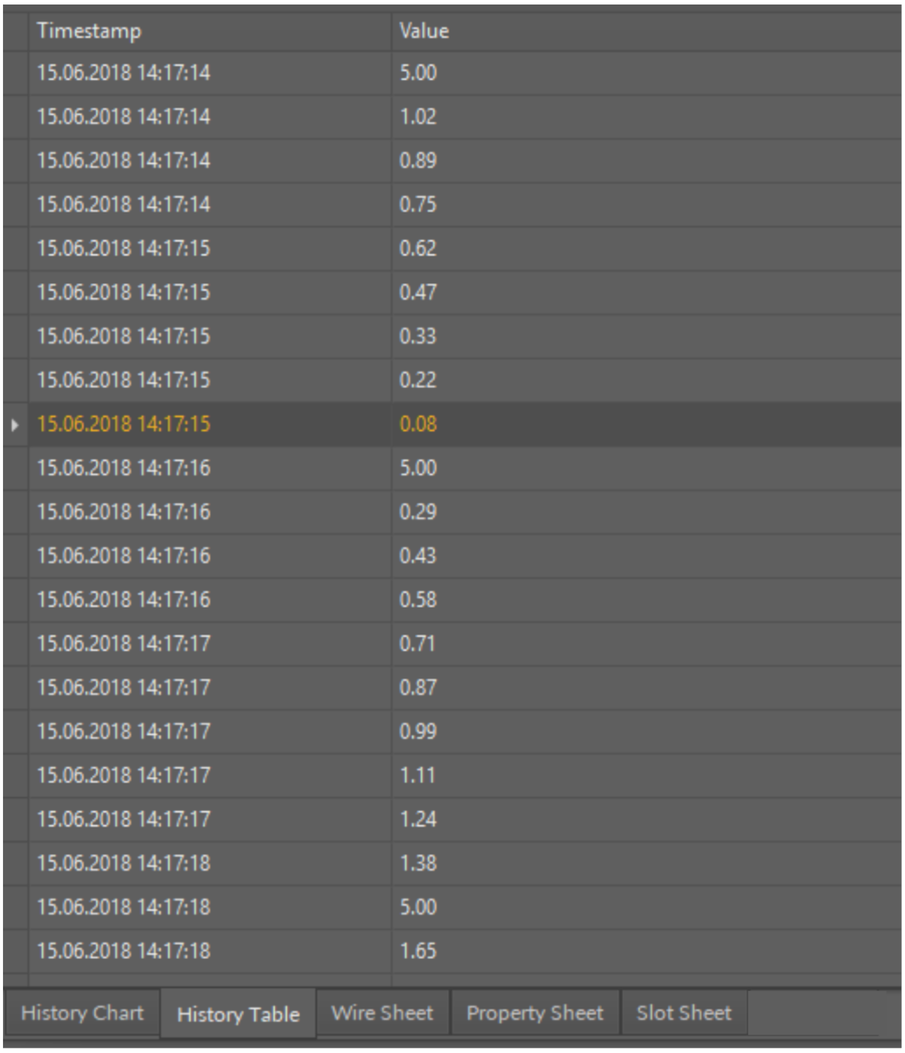
A function of temporary hiding of historical trends is available when working with many trends. In order to temporarily hide the trend, deselect it on the key placed in the upper right corner of the graph.

The remaining options of the context menu allow to return to the default values of zoom with the Reset Zoom function, and to export or print the graph with the Export/Print function.

Initiating the Export/Print function starts a Preview application allowing to print and export the displayed graph. The offered export formats are .pdf, .html, .docx, .xlsx, .bmp, .gif, .jpeg, .png, .tiff, .emf, and .wmf. A detailed description of the function is available in section Exporting and Printing Program–Preview.

10.2.3 History Table View

The History Table view shows all saved samples for one historical component in a tabular form. As in case of the History Chart view, the table can be printed or exported using the Preview program. The Preview program is initiated from the context menu in this view and its description is available in section Exporting and Printing Program–Preview.



Timestamp	Value
15.06.2018 14:17:14	5.00
15.06.2018 14:17:14	1.02
15.06.2018 14:17:14	0.89
15.06.2018 14:17:14	0.75
15.06.2018 14:17:15	0.62
15.06.2018 14:17:15	0.47
15.06.2018 14:17:15	0.33
15.06.2018 14:17:15	0.22
15.06.2018 14:17:15	0.08
15.06.2018 14:17:16	5.00
15.06.2018 14:17:16	0.29
15.06.2018 14:17:16	0.43
15.06.2018 14:17:16	0.58
15.06.2018 14:17:17	0.71
15.06.2018 14:17:17	0.87
15.06.2018 14:17:17	0.99
15.06.2018 14:17:17	1.11
15.06.2018 14:17:17	1.24
15.06.2018 14:17:18	1.38
15.06.2018 14:17:18	5.00
15.06.2018 14:17:18	1.65

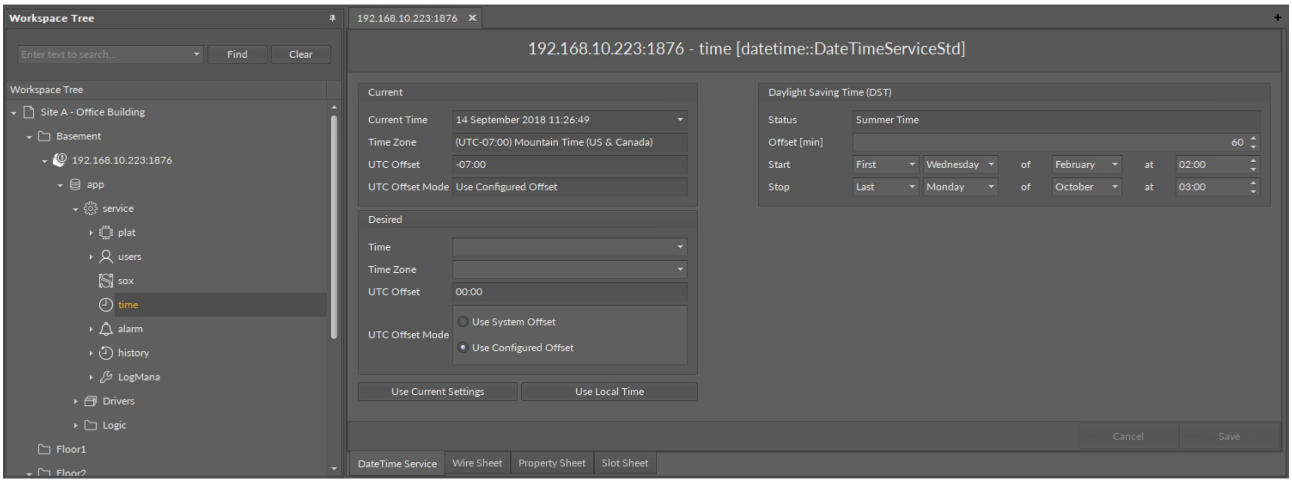
Figure 149. The History Table view

10.3 DateTime Service

The iSMA Tool allows to precisely set the time in the connected controller, including the time zone and a potential change of time connected with daylight saving. For this purpose there is a special DateTimeService view, accessible by double-clicking the time component available at app -> service -> time in the Workspace Tree window.

The DateTimeService shows the following data:

- current time of the controller;
- desired time, which is a setting applied to the controller when saved;
- daylight saving time, which sets an automatic time change on a specific day of the year.



192.168.10.223:1876 - time [datetime::DateTimeServiceStd]

Current

Current Time	14 September 2018 11:26:49
Time Zone	(UTC-07:00) Mountain Time (US & Canada)
UTC Offset	-07:00
UTC Offset Mode	Use Configured Offset

Desired

Time	
Time Zone	
UTC Offset	00:00
UTC Offset Mode	○ Use System Offset ● Use Configured Offset

Use Current Settings | Use Local Time

Daylight Saving Time (DST)

Status	Summer Time					
Offset [min]	60					
Start	First	Wednesday	of	February	at	02:00
Stop	Last	Monday	of	October	at	03:00

Figure 150. The DateTime service

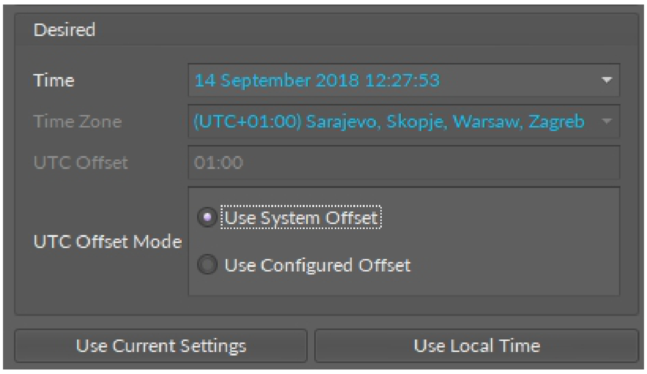
10.3.1 Setting Time and Time Zone

The time and time zone are set in the Desired panel. The new settings are sent to the controller after pressing the Save button in the lower right corner of the view.

To set time in the controller, the time and time zone need to be set up using one of three methods:

- manually enter the time and time zone expanding Time and Time zone fields;
- read the current time and time zone of the controller by pressing the Use Current Settings button, and then edit it;
- read the current time and time zone of the computer operating system in which the iSMA Tool is running by pressing the Use Local Time button, and then edit it.

Using one of the three methods allows to set the time and time zone ready to be sent to the controller, and it is indicated by a blue color of the edited settings (see the figure below).

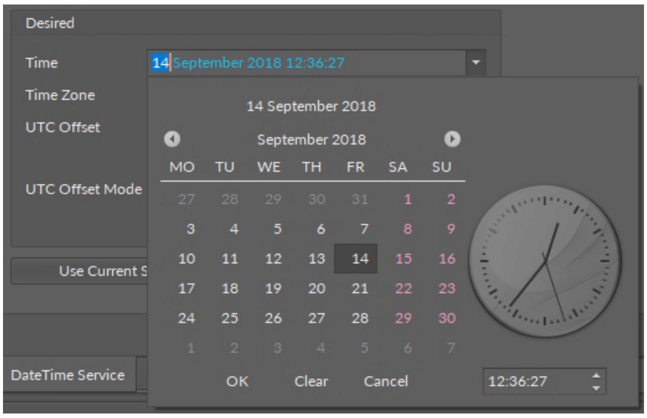


Desired

Time	14 September 2018 12:27:53
Time Zone	(UTC+01:00) Sarajevo, Skopje, Warsaw, Zagreb
UTC Offset	01:00
UTC Offset Mode	● Use System Offset ○ Use Configured Offset

Use Current Settings | Use Local Time

Figure 151. Editing time settings



Desired

Time: 14 September 2018 12:36:27

Time Zone

UTC Offset

UTC Offset Mode

14 September 2018

September 2018

MO	TU	WE	TH	FR	SA	SU
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

12:36:27

Figure 152. Editing time settings

After pressing the Save button, the settings sent to the controller return to their initial colors.

Warning!

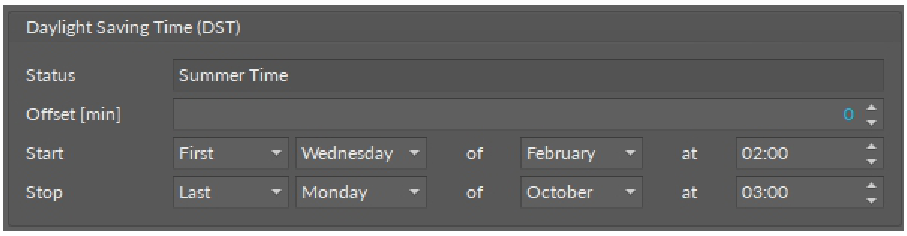
System will not allow setting time before year 2000.

10.3.2 Daylight Saving Time

The Daylight Saving Time (DST) panel on the right side of the DateTimeService view is used for setting the daylight saving time, which is an automatic time change on a specific day of the year.

Warning!

If changing the time due to the daylight saving is not required (e.g., the controller is used in the country that does not use the daylight saving time), 0 should be entered in the Offset field.

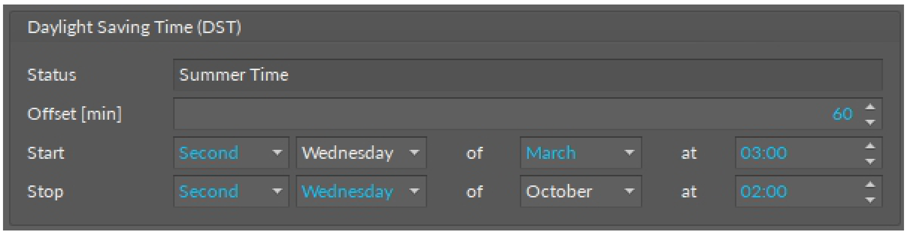


Daylight Saving Time (DST)

Status	Summer Time					
Offset [min]	0					
Start	First	Wednesday	of	February	at	02:00
Stop	Last	Monday	of	October	at	03:00

Figure 153. No Daylight Saving Time settings

When the settings for the beginning and end of daylight saving time are entered, the changed parameters are indicated with a blue color. New settings are sent to the controller with the Save button. When the data are successfully sent to the controller, the color returns to the default one.

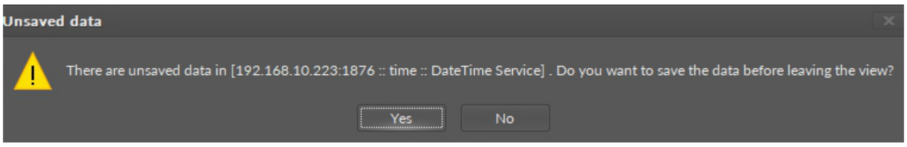


Daylight Saving Time (DST)

Status	Summer Time					
Offset [min]	60					
Start	Second	Wednesday	of	March	at	03:00
Stop	Second	Wednesday	of	October	at	02:00

Figure 154. New settings to DST

Note: In case the DST data are not saved while leaving, the iSMA Tool shows a warning, which should be accepted in order not to lose the entered data.



Unsaved data

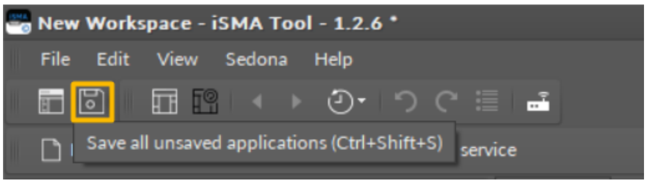
There are unsaved data in [192.168.10.223:1876 :: time :: DateTime Service] . Do you want to save the data before leaving the view?

Yes No

Figure 155. The unsaved data warning

Warning!

Sending the settings to the controller does not mean that the time data are permanently saved in the controller. After sending the data with the Save button, it is necessary to additionally save it in a non-volatile flash memory of the controller by clicking the floppy disk icon on a toolbar in the upper part of the screen.

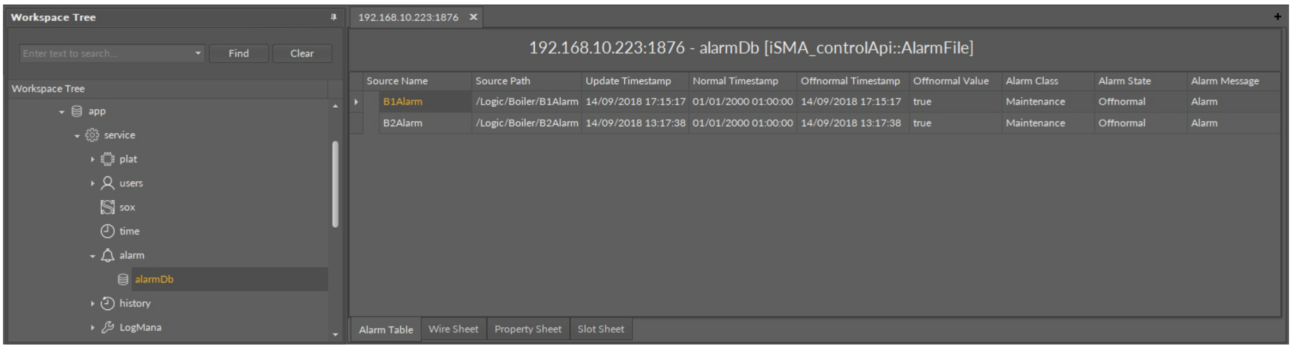


Save all unsaved applications (Ctrl+Shift+S)

Figure 156. Full saving process is confirmed by clicking a floppy disk on a toolbar

10.4 Alarm Service

The iSMA Tool provides a summary table of all the alarms registered in the controller. This table is presented in the Alarm Table view available at app -> service -> alarm -> alarmDb.



192.168.10.223:1876 - alarmDb [iSMA_controlApi::AlarmFile]

Source Name	Source Path	Update Timestamp	Normal Timestamp	Offnormal Timestamp	Offnormal Value	Alarm Class	Alarm State	Alarm Message
B1Alarm	/Logic/Boiler/B1Alarm	14/09/2018 17:15:17	01/01/2000 01:00:00	14/09/2018 17:15:17	true	Maintenance	Offnormal	Alarm
B2Alarm	/Logic/Boiler/B2Alarm	14/09/2018 13:17:38	01/01/2000 01:00:00	14/09/2018 13:17:38	true	Maintenance	Offnormal	Alarm

Figure 157. The Alarm service

Once the Alarm Table view is opened, the iSMA Tool downloads a file with the registered alarms from the controller and displays its contents in a table. The table columns are defined as follows:

- Source Name: shows the name of the component, which generated the alarm;
- Source Path: shows the source component's path;

- Update Timestamp: shows the time of the last update;
- Normal Timestamp: shows the time of switching to normal state after the alarm;
- Offnormal Timestamp: shows the time of switching to alarm state;
- Offnormal Value: shows the alarm value;
- Alarm Class: shows the alarm class;
- Alarm State: shows the alarm status;
- Alarm Message: shows the alarm message.

Warning!

The Alarm view is a non-editable view and its contents are read from the alarm file only while the view is loading.

11 Schedules

In the iSMA Tool there are two components that allow to set schedules for specific events–the BooleanScheduleWeekly and NumericScheduleWeekly. Both offer a Schedule Weekly view, which facilitates setting a schedule, and a Special Events view, which allows to add special events to the schedule.

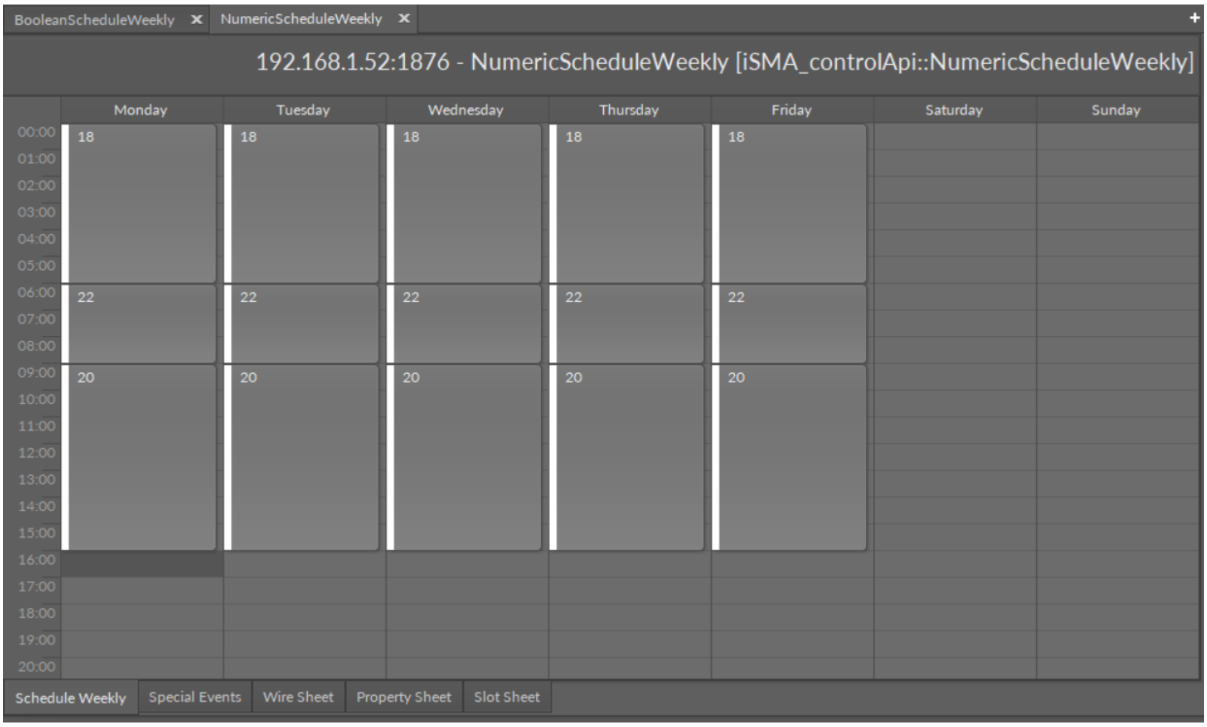


Figure 158. The Schedule Weekly view

In order to open the Schedule Weekly view, double-click either one of the scheduling components. The Schedule Weekly view displays a one week view, each day divided by hours.

To set a schedule, first, set the required time interval. If the required time interval lasts for an hour or less, double-click on the required day and time, open the context menu, choose Add Event option, and enter the required value and time limit in the displayed dialog window. If the required time is longer than an hour, click and hold the click on the beginning of the required time interval and drag until the end of it. Open the context menu, choose Add Event option, and set the required value in the displayed dialog window.

In case no specific value is set for a given time interval, the default value is applied.

The example from the figure 141 shows the schedule, which may be read as a room temperature profile. From Monday to Friday, from 00:00 AM to 06:00 AM the set value is 18, and then it changes from 06:00 AM to 09:00 AM to 22. From 09:00 AM to 04:00 PM the set value is 20. After 04:00 PM the default value is applied.

Editing of the schedule involves selecting the required time interval and using the context menu. The context menu offers the following options:

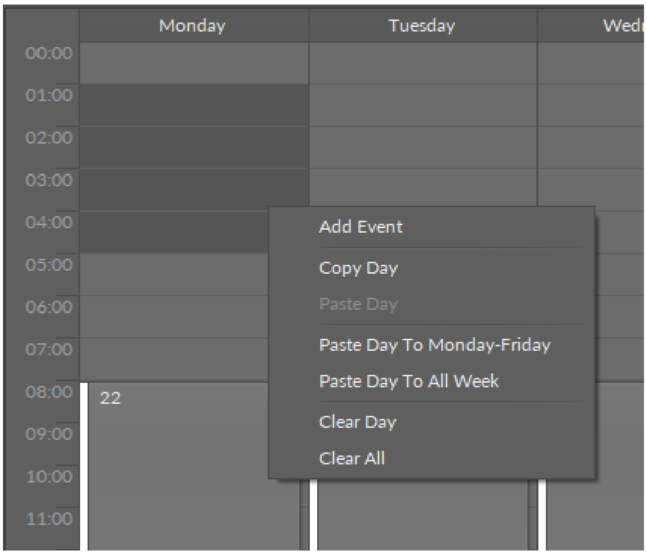
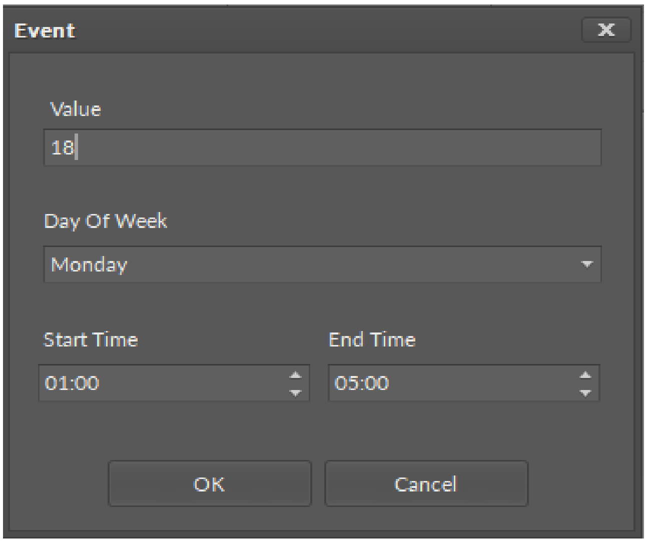


Figure 159. A context menu in the Schedule Weekly view

- Add Event: allows to create a new event, which means setting a required value to be applied in the define time. The Add Event option opens a dialog window:


Event

Value: 18

Day Of Week: Monday

Start Time: 01:00

End Time: 05:00

Figure 160. The Add Event dialog window

The Value field allows to set a required value, depending on the scheduling component type, it is a numeric or Boolean value.

The Day Of Week field allows to adjust the starting day of the selected time interval.

The Start Time and End Time fields allow to adjust the selected time interval.

The event is created after confirming with the OK button in the dialog window. Now, on Monday, from 01:00 AM to 05:00 AM the setting will be 18.

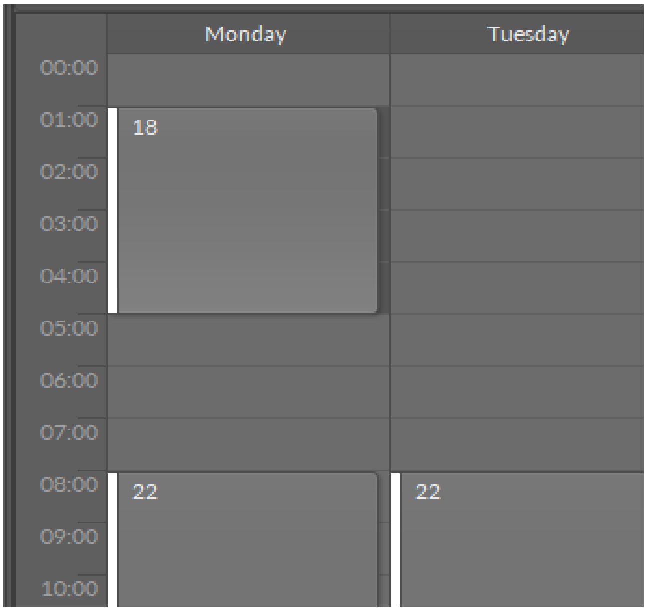


Figure 161. New setting in a schedule

- Copy Day: allows to copy all events of a day;
- Paste Day: allows to overwrite the selected day with the events copied from another day (Copy Day);
- Paste Day To Monday-Friday: allows to copy and overwrite all the events of selected day to all days from Monday to Friday;
- Paste Day To All Week: similar to the Paste Day To Monday-Friday option, but in this instance all days of the week will be overwritten;
- Clear Day: allows to remove all events of a day;
- Clear All: allows to remove all events from the schedule, regardless of which day the option was initiated on.

To edit of an already existing event:

- drag the event to a new time in a calendar;
- pull the upper or bottom border of the event, changing its beginning or ending hours

Note: The events cannot overlap during one day.

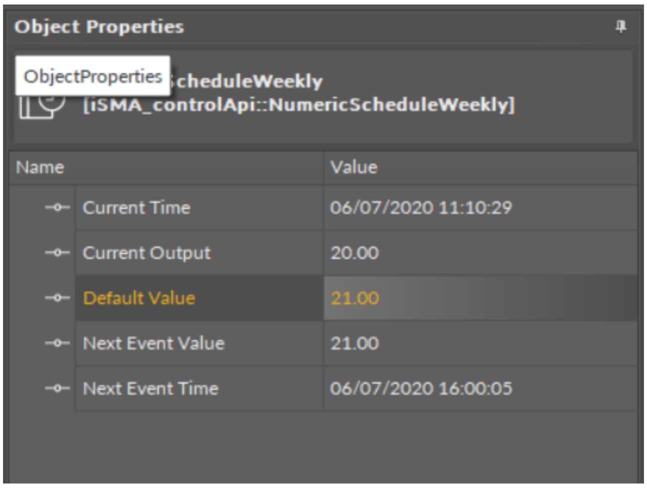
- double-click the event;
- open the event's context menu and choose the Edit Event option.

The selected event may be removed using Delete button on the keyboard or choosing the Delete Event option from the context menu.

Note: All changes in the schedule are sent directly to the controller and are immediately taken into account in the application's logic. It means that removing any event may result in a change of the application's behavior due to the change of the schedule output parameter value.

While using the Schedule Weekly view, additional data are displayed in the Object Properties window:

- Current Time: shows the current time;
- Current Output: shows the current output value;
- Default Value: allows to set a default value, which is used in case no event is planned;
- Next Event Value: shows the output value of the next scheduled event;
- Next Event Time: shows the time when the next output value will be applied.



cheduleWeekly
[iSMA_controlApi::NumericScheduleWeekly]

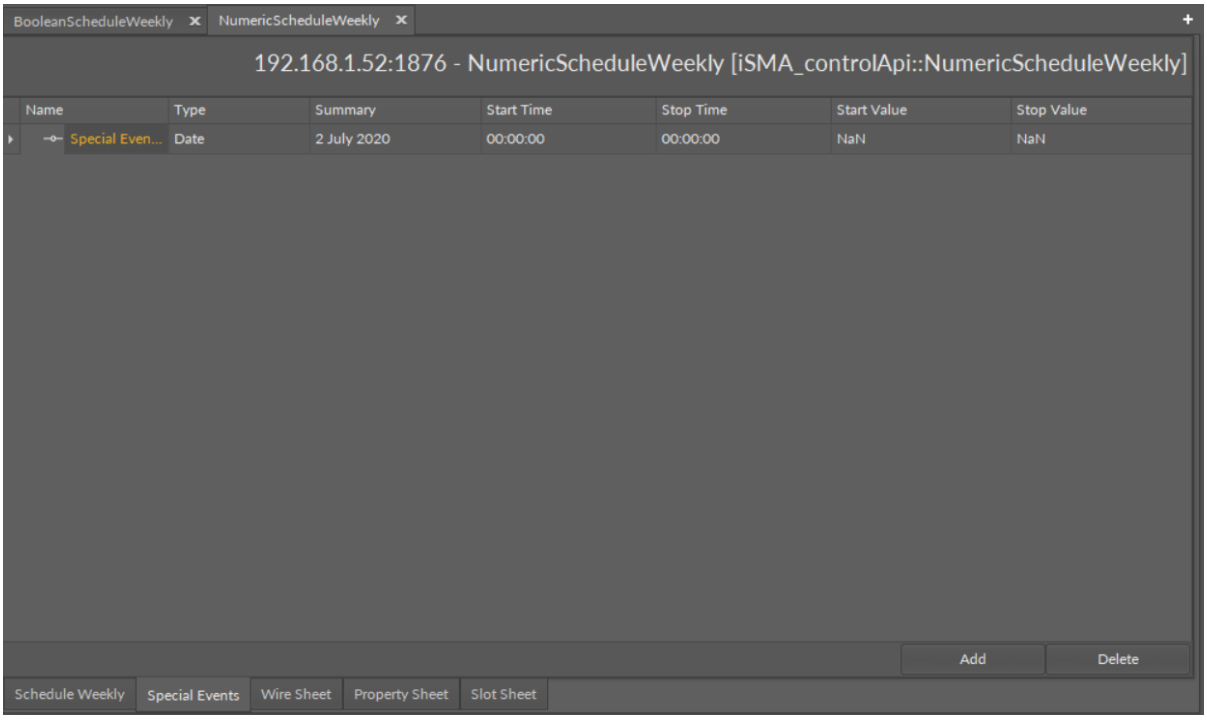
Name	Value
Current Time	06/07/2020 11:10:29
Current Output	20.00
Default Value	21.00
Next Event Value	21.00
Next Event Time	06/07/2020 16:00:05

Figure 162. The Object Properties view for scheduling components

Note: All settings, which are available in the Schedule Weekly view and the Object Properties window, can also be edited in the Property Sheet view.

11.1 Special Events

Apart from the Schedule Weekly view, the scheduling components have also the Special Events view available.



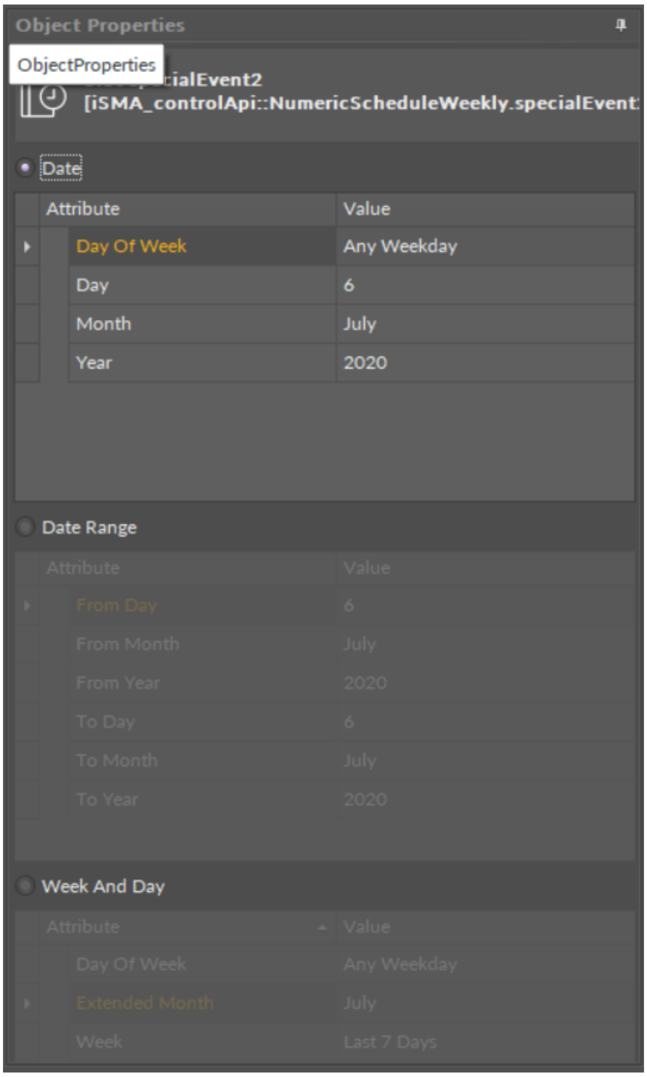
192.168.1.52:1876 - NumericScheduleWeekly [iSMA_controlApi::NumericScheduleWeekly]

Name	Type	Summary	Start Time	Stop Time	Start Value	Stop Value
Special Even...	Date	2 July 2020	00:00:00	00:00:00	NaN	NaN

Figure 163. The Special Events view

The Special Events tab allows to set individual events, which are not part of a weekly schedule. These events are not displayed in the Schedule Weekly view. They are added or deleted using the Add or Delete buttons at the bottom of the tab. The settings available for a special event are the following:

- Type: the setting edited in the Object Properties window; the special event can be set either for a specific date, the date range, or a specific week and day (this option allows also to select month which the special event will take place in);



cialEvent2
[iSMA_controlApi::NumericScheduleWeekly.specialEvent

● Date

Attribute	Value
Day Of Week	Any Weekday
Day	6
Month	July
Year	2020

○ Date Range

Attribute	Value
From Day	6
From Month	July
From Year	2020
To Day	6
To Month	July
To Year	2020

○ Week And Day

Attribute	Value
Day Of Week	Any Weekday
Extended Month	July
Week	Last 7 Days

Figure 164. Special Event in the Object Properties window

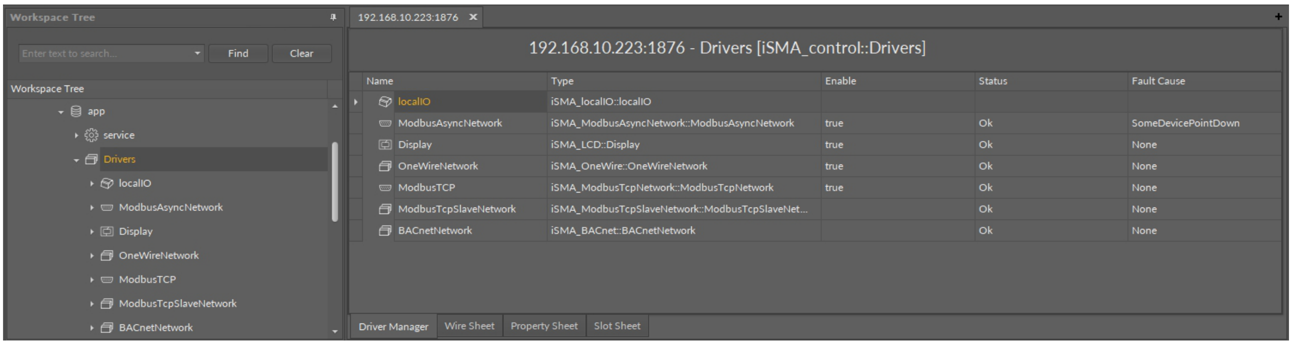
- Start Time: the starting time for the special event;
- Stop Time: the ending time of the special event;
- Start Value: the starting value of the special event;
- Stop Value: the ending value of the special event.

12 Drivers Views

The iSMA Tool has specially designed tabular drivers views to make the work of adding and maintaining the devices working in the network and exchanging data among them as easy as possible. Predefined drivers views will be discussed based on an exemplary complex configuration of many networks and devices.

Tabular drivers views may be divided into 3 hierarchical segments:

- First segment is a table with the view of all drivers, that is networks. It is called Driver Manager. For example, the table shows the status of, among others, the following networks: localIO, ModbusAsync, OneWire, ModbusTCP, BACnet.

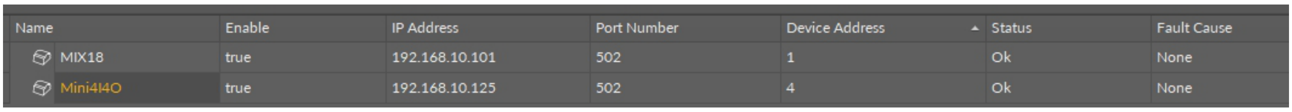


192.168.10.223:1876 - Drivers [iSMA_control::Drivers]

Name	Type	Enable	Status	Fault Cause
localIO	iSMA_localIO::localIO			
ModbusAsyncNetwork	iSMA_ModbusAsyncNetwork::ModbusAsyncNetwork	true	Ok	SomeDevicePointDown
Display	iSMA_LCD::Display	true	Ok	None
OneWireNetwork	iSMA_OneWire::OneWireNetwork	true	Ok	None
ModbusTCP	iSMA_ModbusTcpNetwork::ModbusTcpNetwork	true	Ok	None
ModbusTcpSlaveNetwork	iSMA_ModbusTcpSlaveNetwork::ModbusTcpSlaveNet...		Ok	None
BACnetNetwork	iSMA_BACnet::BACnetNetwork		Ok	None

Figure 165. The first segment of tabular drivers views

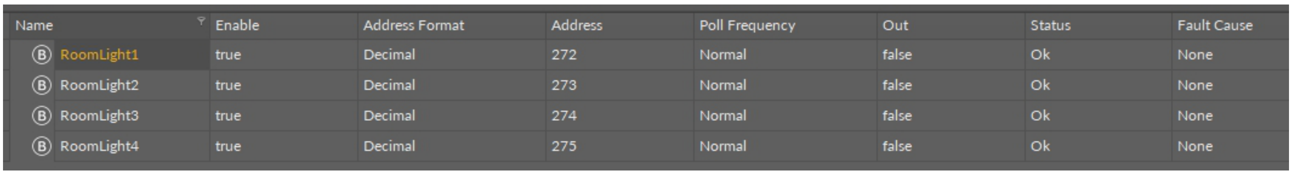
- Second segment shows the devices included in the network selected from the first segment. In the below example, the devices are ‘MIX18’ and ‘Mini4I4O’ for the network (driver)–ModbusTCP. The name of this view depends on the type of the chosen network, here it is the Modbus TCP Device Manager view.



Name	Enable	IP Address	Port Number	Device Address	Status	Fault Cause
MIX18	true	192.168.10.101	502	1	Ok	None
Mini4I4O	true	192.168.10.125	502	4	Ok	None

Figure 166. The second segment of tabular drivers views

- Third segment is a list of network points, which are variables allowing to read/record particular data from a particular device. In the below example, the digital points are RoomLight1-4, and they are read by the iSMA-B-4I4O device. The name of the views showing network points depends on the name of the device the data comes from, e.g., Modbus TCP Point Manager view.



Name	Enable	Address Format	Address	Poll Frequency	Out	Status	Fault Cause
Ⓑ RoomLight1	true	Decimal	272	Normal	false	Ok	None
Ⓑ RoomLight2	true	Decimal	273	Normal	false	Ok	None
Ⓑ RoomLight3	true	Decimal	274	Normal	false	Ok	None
Ⓑ RoomLight4	true	Decimal	275	Normal	false	Ok	None

Figure 167. The third segment of tabular drivers views

Generally, the three drivers views’ segments described above may graphically be presented as follows:

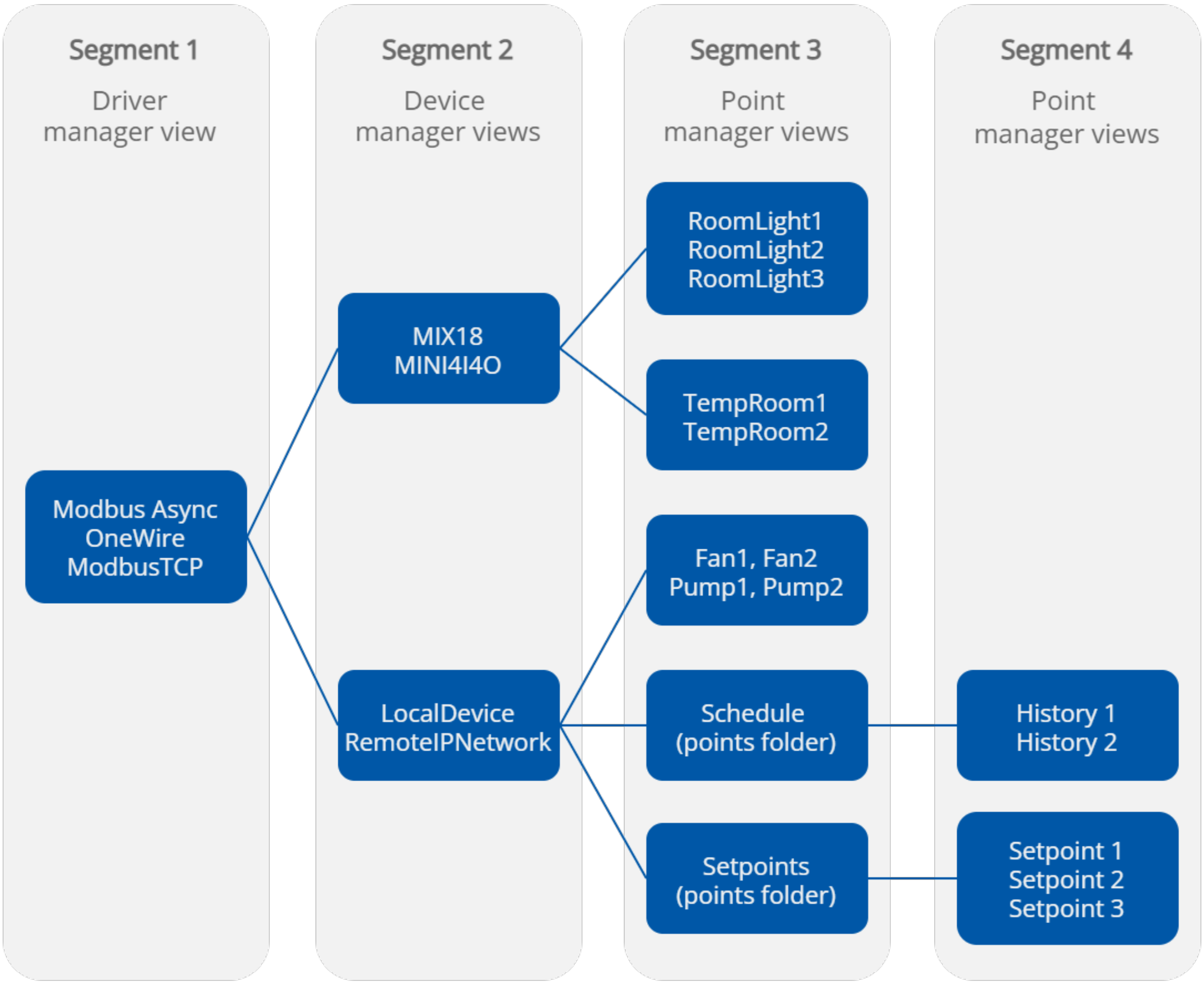


Figure 168. Driver views segments graphically

As depicted in the above diagram, network points may be grouped in network folders (Points Folder) and nested any number of times, which allows grouping and ordering large quantities of network points under a particular device (here, Schedule and Setpoints are folders grouping points History1,2 and Setpoint1,2,3, respectively).

The views construction of particular segments is very similar, so one type of table for each of the segments may display all the necessary data.

The next part focuses on how networks should be built based on devices distributed among Kits and points available in the iSMA Tool.

The further part focuses on how the XML configuration files are built and how the inbuilt tabular views for drivers views were defined. That knowledge may be used to build custom tabular views to expand drivers views with custom networks, devices, points, or to create completely new views for other application zones.

12.1 Using Drivers Views

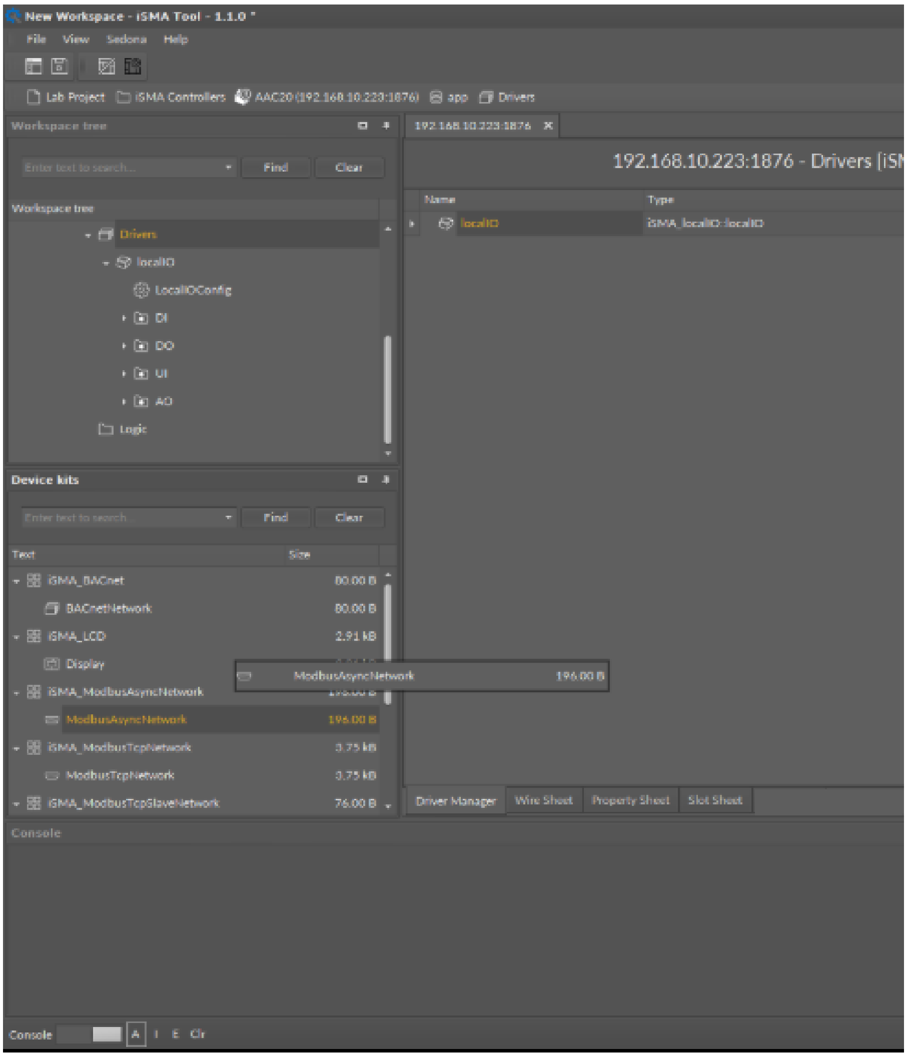
This section shows how to add a network, devices and points, using built-in tabular views divided into segments.

12.1.1 Segment 1

To add a new network (driver) to the application:

- open a proper Driver Manager view available at app -> Drivers in the Workspace Tree;
- drag the required driver (component) representing the proper network from the Device Kits window;

Note: To make working with drivers views (but also device and point views) easier, the iSMA Tool filters components available to use in the Device Kits window, leaving only the ones, which can be added to the particular view. The figure below shows the open Driver Manager view with a list of available components in the Device Kits window. The list of components has been limited to these, which define networks, e.g., ModbusAsyncNetwork, ModbusTCPNetwork, ModbusTcpSlaveNetwork, OneWireNetwork, localIO, etc. The figure below shows the process of adding a driver ModbusAsyncNetwork to the Driver Manager view.

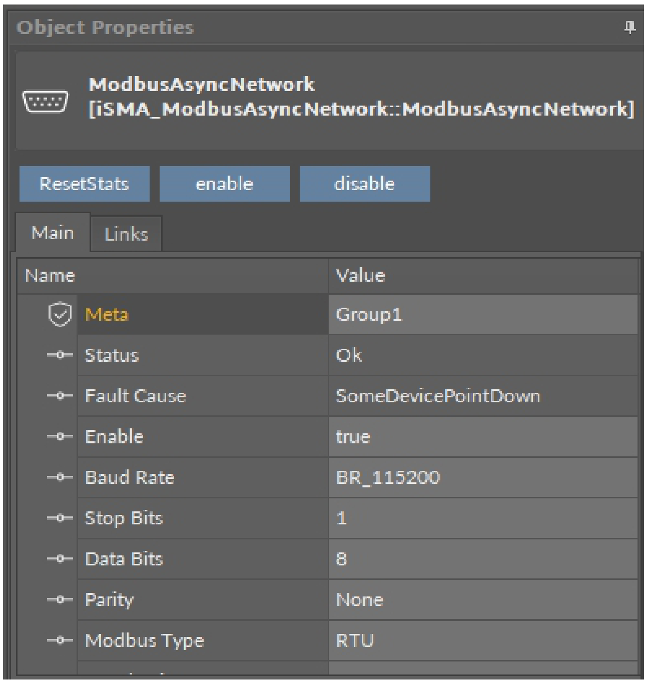


Name	Type
localIO	iSMA_localIO::localIO

Text	Size
iSMA_BACnet	80.00 B
BACnetNetwork	80.00 B
iSMA_LCD	2.91 kB
Display	[illegible]
iSMA_ModbusAsyncNetwork	[illegible]
ModbusAsyncNetwork	196.00 B
iSMA_ModbusTcpNetwork	3.75 kB
ModbusTcpNetwork	3.75 kB
iSMA_ModbusTcpSlaveNetwork	76.00 B

Figure 169. Adding a component to the Drivers view

- adjust the necessary settings for the added network–it is best to use the Object Properties window, which shows parameters of the added driver after it has been selected, see the figure below. A basic thing to set in the beginning is to enable the network using the Enable action in the Object properties window.



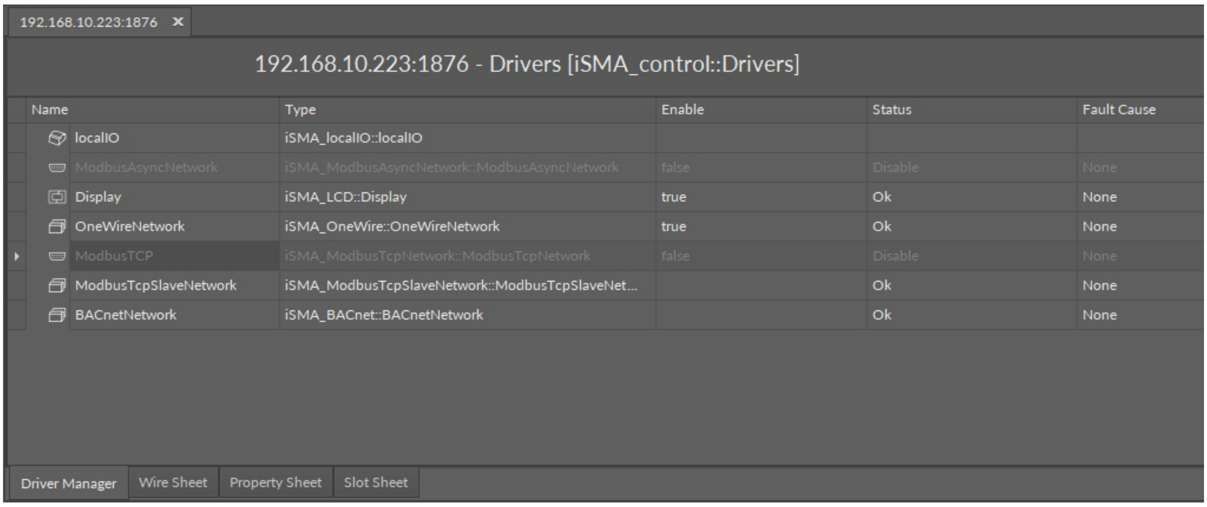
ModbusAsyncNetwork
[iSMA_ModbusAsyncNetwork::ModbusAsyncNetwork]

Name	Value
Meta	Group1
Status	Ok
Fault Cause	SomeDevicePointDown
Enable	true
Baud Rate	BR_115200
Stop Bits	1
Data Bits	8
Parity	None
Modbus Type	RTU

Figure 170. The Object Properties of an added network

Warning!

Inactive disabled drivers are marked with gray color, see the figure below.



192.168.10.223:1876 - Drivers [iSMA_control::Drivers]

Name	Type	Enable	Status	Fault Cause
localIO	iSMA_localIO::localIO			
ModbusAsyncNetwork	iSMA_ModbusAsyncNetwork::ModbusAsyncNetwork	false	Disable	None
Display	iSMA_LCD::Display	true	Ok	None
OneWireNetwork	iSMA_OneWire::OneWireNetwork	true	Ok	None
ModbusTCP	iSMA_ModbusTcpNetwork::ModbusTcpNetwork	false	Disable	None
ModbusTcpSlaveNetwork	iSMA_ModbusTcpSlaveNetwork::ModbusTcpSlaveNet...		Ok	None
BACnetNetwork	iSMA_BACnet::BACnetNetwork		Ok	None

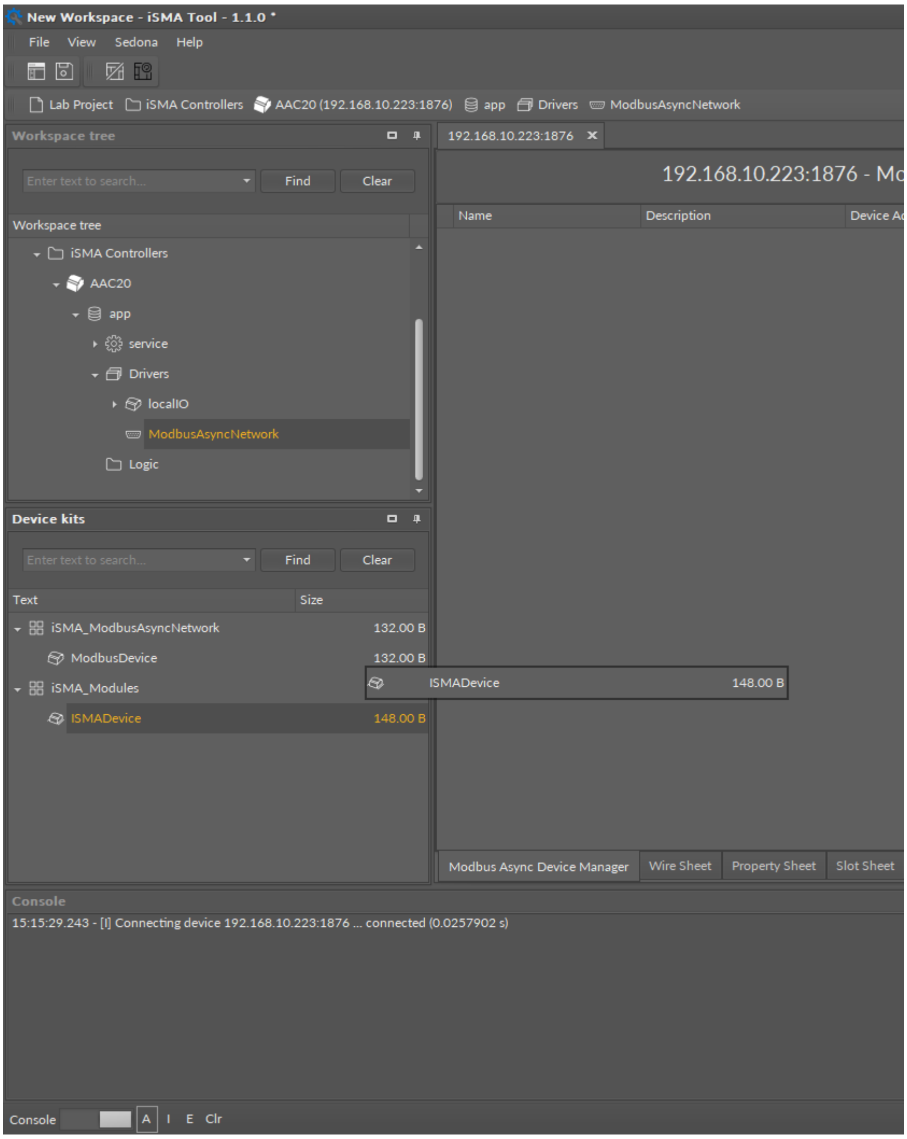
Figure 171. Disabled network view

12.1.2 Segment 2

Next, it is necessary to add the devices to the ModbusAsyncNetwork, which are part of this network:

- double-click the added network (driver) and open the Modbus Async Device Manager view;
- drag the components representing device types installed in this network into the view. The figure below shows dragging the component representing the iSMADevice type of device.

Note: To make working with device views easier, the iSMA Tool filters devices, which support the selected communications network (ModbusAsync); in the example below the devices are: ModbusDevice and iSMADevice.

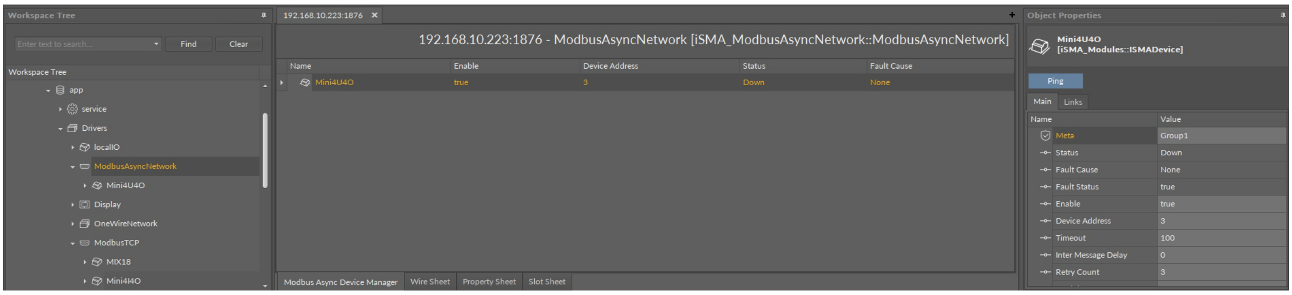


Text	Size
iSMA_ModbusAsyncNetwork	132.00 B
ModbusDevice	132.00 B
iSMA_Modules	
ISMADevice	148.00 B

```
15:15:29.243 - [I] Connecting device 192.168.10.223:1876 ... connected (0.0257902 s)
```

Figure 172. Adding a device to the network

- configure the device settings in the Object Properties window.



192.168.10.223:1876 - ModbusAsyncNetwork [iSMA_ModbusAsyncNetwork::ModbusAsyncNetwork]

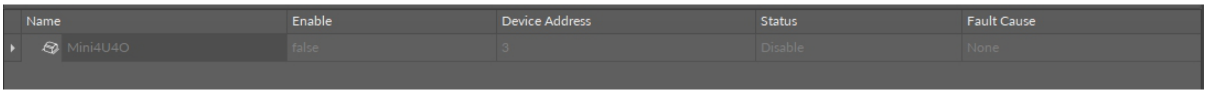
Name	Enable	Device Address	Status	Fault Cause
Mini4U4O	true	3	Down	None

Name	Value
Meta	Group1
Status	Down
Fault Cause	None
Fault Status	true
Enable	true
Device Address	3
Timeout	100
Inter Message Delay	0
Retry Count	3

Figure 173. COnfigure the device in the Object Properties window

Warning!

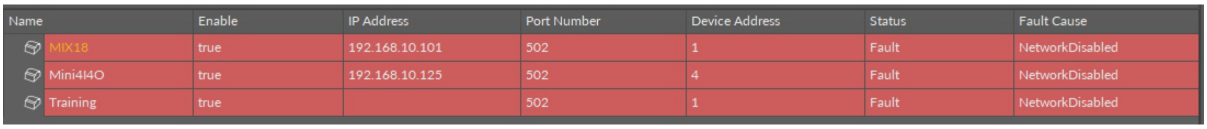
Inactive devices are indicated by gray color, as shown in the figure below. To activate a device, the Enabled slot needs to be set to true using the Object Properties window.



Name	Enable	Device Address	Status	Fault Cause
Mini4U4O	false	3	Disable	None

Figure 174. Disabled device view

Note: Red color indicates faulty devices with Fault status, see the figure below.



Name	Enable	IP Address	Port Number	Device Address	Status	Fault Cause
MIX18	true	192.168.10.101	502	1	Fault	NetworkDisabled
Mini4I4O	true	192.168.10.125	502	4	Fault	NetworkDisabled
Training	true		502	1	Fault	NetworkDisabled

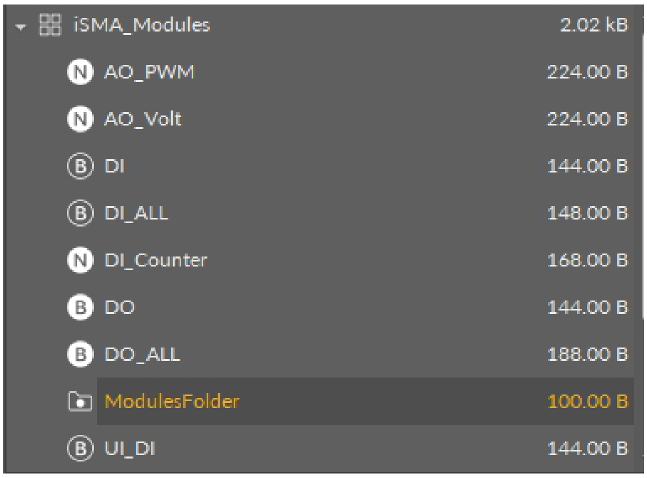
Figure 175. Faulty devices view

12.1.3 Segment 3

Next, add network points to the defined device, so the application in the controller will be able to read and record data from/to an external device:

- open the Modbus Async iSMA Module Point Manager view double-clicking the freshly added device;

This time as well, the component list in Device Kits is limited to those, which cooperate with the selected device. Besides typical read and write points (here components with N and B icons), there is also a component of another type, the ModulesFolder, see the selected folder in the figure below.



iSMA_Modules	2.02 kB
N AO_PWM	224.00 B
N AO_Volt	224.00 B
B DI	144.00 B
B DI_ALL	148.00 B
N DI_Counter	168.00 B
B DO	144.00 B
B DO_ALL	188.00 B
ModulesFolder	100.00 B
B UI_DI	144.00 B

Figure 176. The ModulesFolder container

The ModulesFolder is a container (folder) allowing to group the network points. It may be nested in the third segment views any number of times. The figure below shows the process of adding the UI_Temp network point to the iSMA-B-4U4O device.

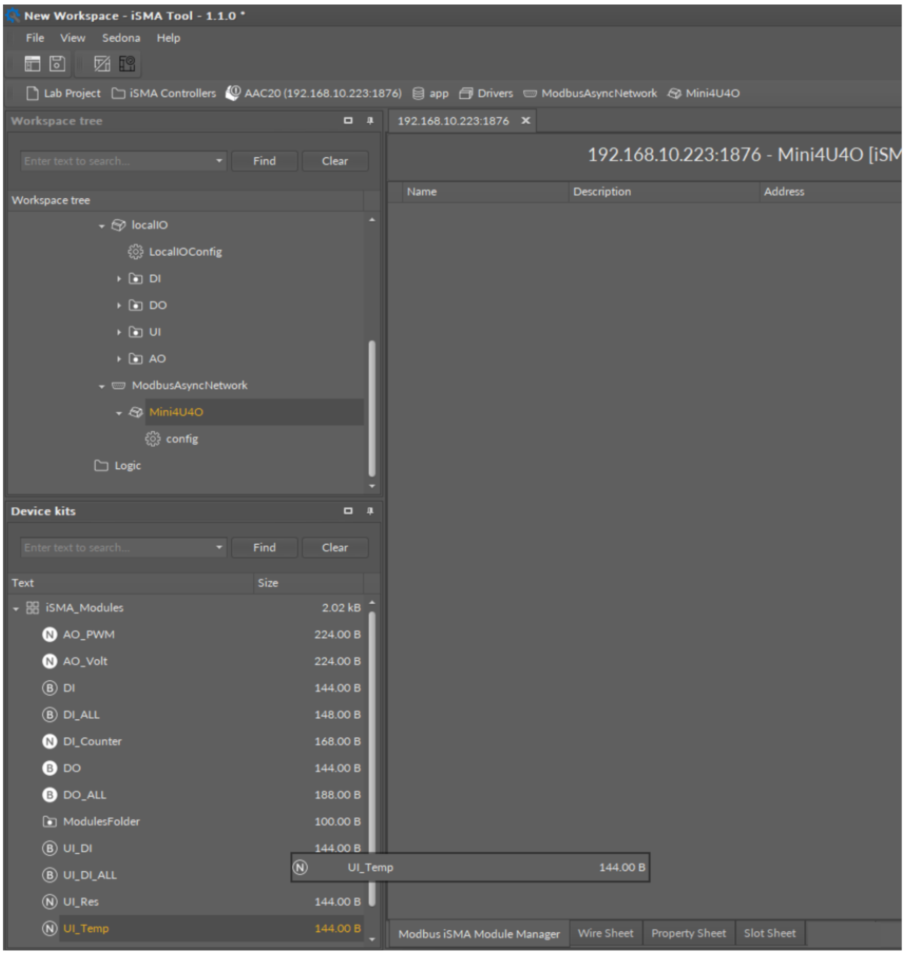


Figure 177. Adding a network point to the device

Warning!

To determine which device the particular network point belongs to, or which network it belongs to, the access path (figure below) is displayed above the Main window in the upper part of the screen. This dependency is also visible in the Workspace Tree.

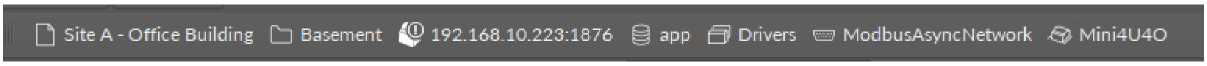
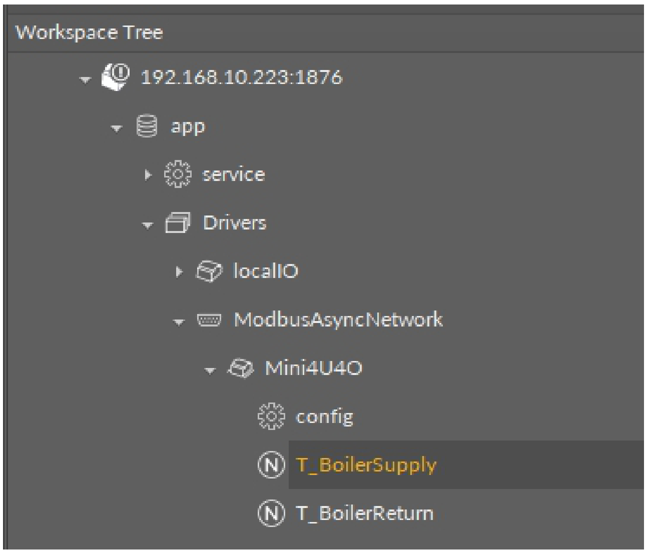


Figure 178. Device and network points dependency



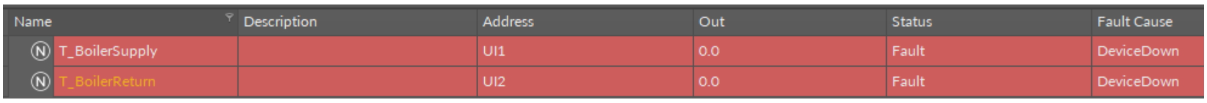
Workspace Tree
- 192.168.10.223:1876
  - app
    - service
    - Drivers
      - localIO
      - ModbusAsyncNetwork
        - Mini4U4O
          - config
          - T_BoilerSupply
          - T_BoilerReturn

Figure 179. Device and network points dependency

- configure the parameters required for each of network point separately, e.g., the address, in the Object Properties window.

Warning!

Red color indicates faulty points with 'Fault' status, see the figure below.



Name	Description	Address	Out	Status	Fault Cause
T_BoilerSupply		UI1	0.0	Fault	DeviceDown
T_BoilerReturn		UI2	0.0	Fault	DeviceDown

Figure 180. Faulty network points view

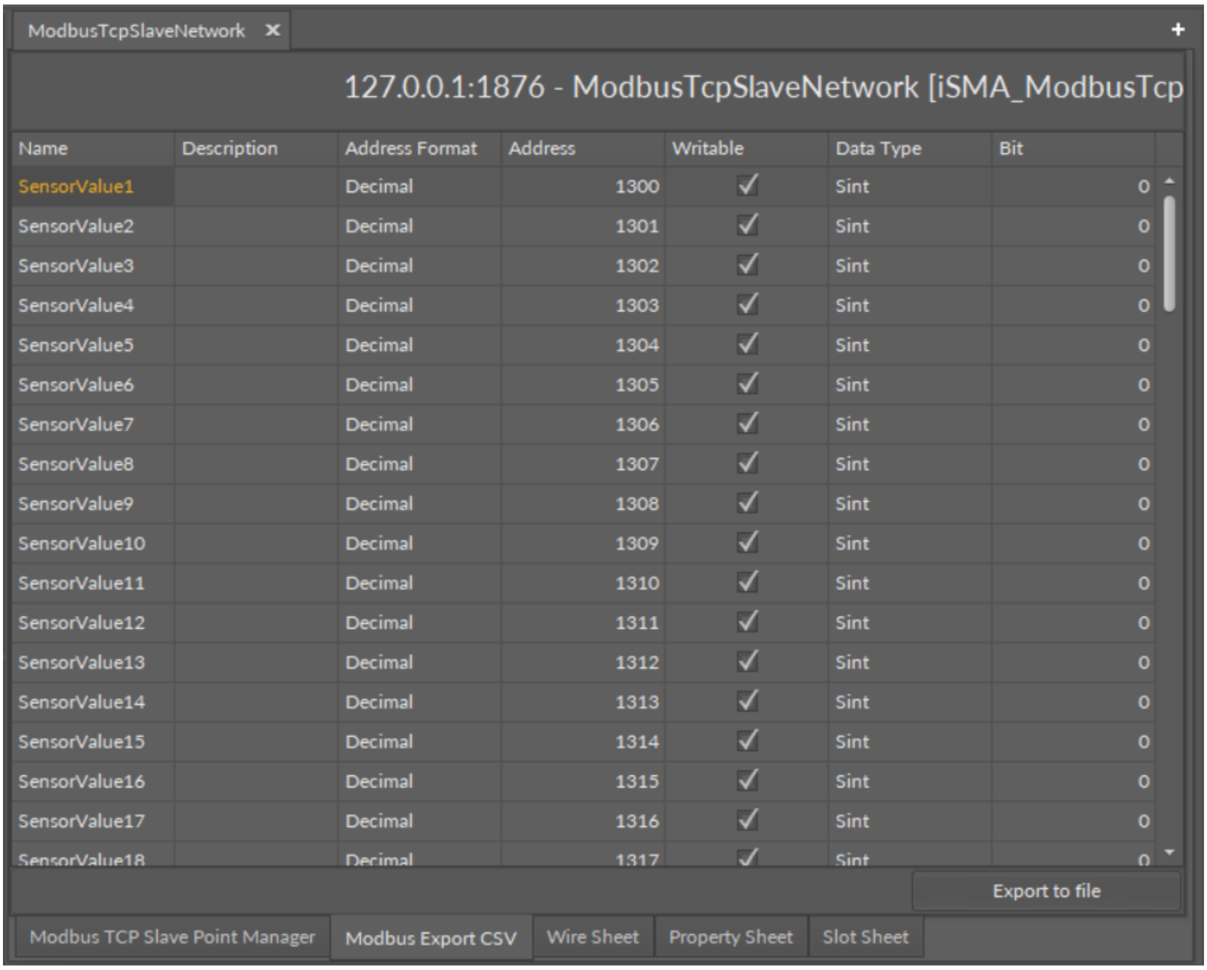
After successfully configuring the network points parameters, its device, and network, which the device works in, the current value will be read by the network point.

The process described above needs to be performed repeatedly for every network, every device in network, and all required network points.

12.1.4 Modbus CSV Export

For Modbus TCP and Modbus Async slave networks, an additional view is available–the Modbus Export CSV view. The view lists all points in the network with the following parameters and allows to export the list to a CSV file:

- name;
- description;
- address format;
- address;
- information if the point writable or not;
- data type;
- bit number.



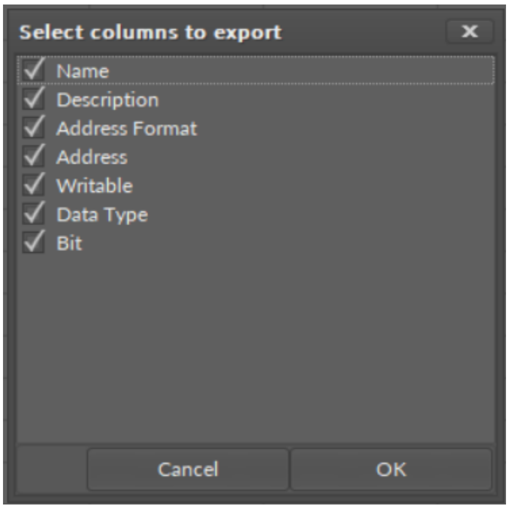
ModbusTcpSlaveNetwork

127.0.0.1:1876 - ModbusTcpSlaveNetwork [iSMA_ModbusTcp

Name	Description	Address Format	Address	Writable	Data Type	Bit
SensorValue1		Decimal	1300	✓	Sint	0
SensorValue2		Decimal	1301	✓	Sint	0
SensorValue3		Decimal	1302	✓	Sint	0
SensorValue4		Decimal	1303	✓	Sint	0
SensorValue5		Decimal	1304	✓	Sint	0
SensorValue6		Decimal	1305	✓	Sint	0
SensorValue7		Decimal	1306	✓	Sint	0
SensorValue8		Decimal	1307	✓	Sint	0
SensorValue9		Decimal	1308	✓	Sint	0
SensorValue10		Decimal	1309	✓	Sint	0
SensorValue11		Decimal	1310	✓	Sint	0
SensorValue12		Decimal	1311	✓	Sint	0
SensorValue13		Decimal	1312	✓	Sint	0
SensorValue14		Decimal	1313	✓	Sint	0
SensorValue15		Decimal	1314	✓	Sint	0
SensorValue16		Decimal	1315	✓	Sint	0
SensorValue17		Decimal	1316	✓	Sint	0
SensorValue18		Decimal	1317	✓	Sint	0

Figure 181. The Modbus Export CSV view

To export the list, click the Export to file button. Exported file can include all data or specific columns:



Select columns to export

- ☑ Name
- ☑ Description
- ☑ Address Format
- ☑ Address
- ☑ Writable
- ☑ Data Type
- ☑ Bit

Figure 182. Selecting columns to export

Press OK to confirm, and the file is downloaded automatically.

13 Custom Tabular View

The iSMA Tool allows each user to define individual tabular views for any component by defining an individual .xml file.

In such case the Tabular view is available beside main views of each component, which are the Wire Sheet, Property Sheet, and Slot Sheet.

The Tabular view filters subordinate components (children) making it much easier to work with complex applications composed of many components (children) of different types.

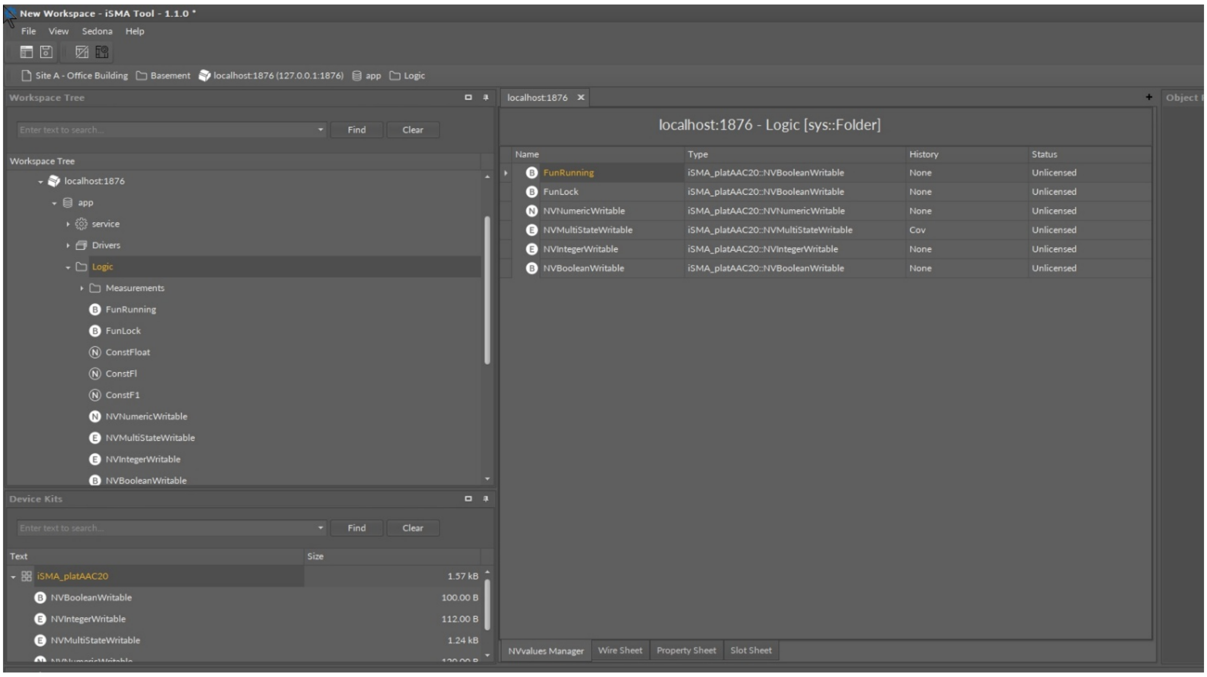
In a situation when a folder-type component (parent) has many different components, under it, including, for example, NV components, the Tabular view facilitates searching for the required components. Without the Tabular view it would be necessary to identify required components among many other components, and view their values in the Object Properties window. By defining one .xml file properly it is possible to create an individual Tabular view for each type of component in the iSMA Tool (it will list only the required type of components). For example, such view would be filter all NV components.

iSMA CONTROLLI wants the user to be able to create tabular summaries of components according to his individual needs, without the need to delegate it to iSMA CONTROLLI.

Each view created that way will require following characteristics to be defined:

- Tabular view name;
- Types of components for which the tabular view will be displayed;
- Types of components, which will be displayed in tabular view along with their common slots.

The figure below shows an example of a tabular view filtering NV components. While working with a view defined this way (here named 'NVValuesManager'), the iSMA Tool limits the number of components available in the Device Kits window, making it easier to add proper components to the view. After switching to other view (choosing a tab for other views), the Device Kits window will return to normal working mode displaying all components available in the device.



localhost:1876 - Logic [sys::Folder]

Name	Type	History	Status
FunRunning	iSMA_platAAC20::NVBooleanWritable	None	Unlicensed
FunLock	iSMA_platAAC20::NVBooleanWritable	None	Unlicensed
NVNumericWritable	iSMA_platAAC20::NVNumericWritable	None	Unlicensed
NVMultiStateWritable	iSMA_platAAC20::NVMultiStateWritable	Cov	Unlicensed
NVIntegerWritable	iSMA_platAAC20::NVIntegerWritable	None	Unlicensed
NVBooleanWritable	iSMA_platAAC20::NVBooleanWritable	None	Unlicensed

Text	Size
iSMA_platAAC20	1.57 kB
NVBooleanWritable	100.00 B
NVIntegerWritable	112.00 B
NVMultiStateWritable	1.24 kB

Figure 183. Custom table view

The iSMA Tool provides properly configured tabular views for all the devices from the iSMA family, which are described in detail in the next chapter.

13.1 Creating Custom Tabular View

The iSMA Tool allows to define the view contents as well as compatible (fitting) components through configuring a proper .xml file. This approach allows the iSMA Tool user to build required views for any type of component in the application.

The .xml configuration files need to be placed in the iSMA Tool main folder under /home/CustomTableViews folder.

By default the folder includes the following files created for tabular drivers views:

- one drivers.xml file–defining all drivers (networks) available in the iSMA Tool;
- [driver name]DeviceManager.xml or [driver name]NetworkManager.xml files–defining the look for a particular driver (device) view;
- [Driver name]PointManager.xml files–defining the look of network points for a particular device.

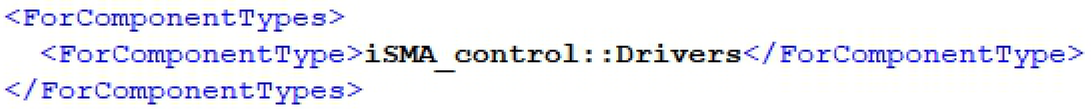
The .xml files have the following tags, which meaning is necessary for proper construction of any tabular views:

- <ViewName>Driver Manager</ViewName>–defines the name of the view connected to the settings in remaining tags;

```
<ViewName>Driver Manager</ViewName>
```

Figure 184. Creating custom table view

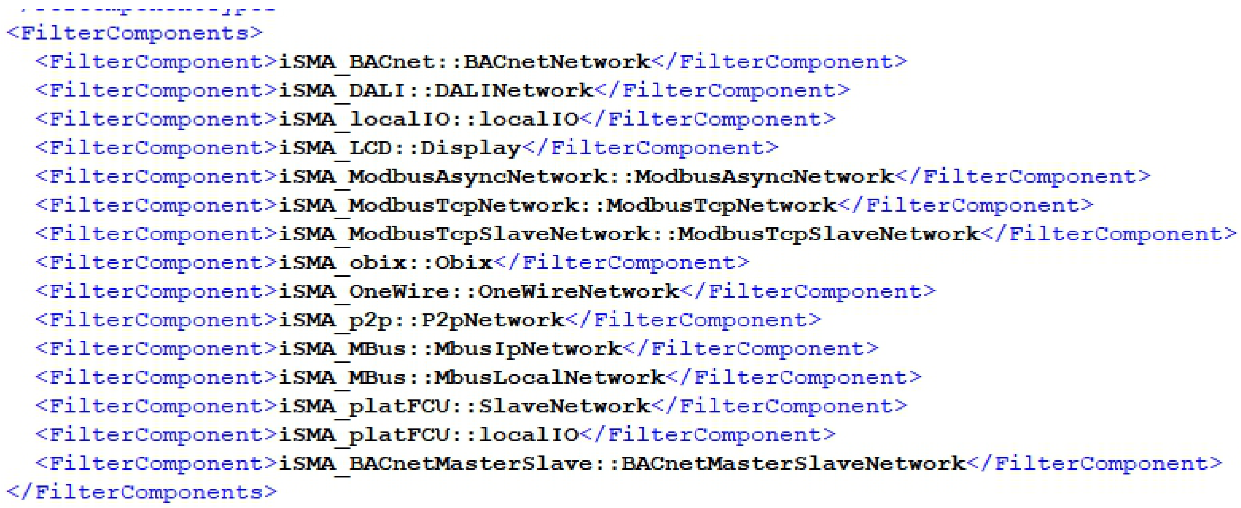
- <ForcomponentTypes>–defines the names of components, for which the particular view will be used. This means that for components defined within this tag, one of the assigned views will be a view defined by <ViewName> tag;



```
<ForComponentTypes>
  <ForComponentType>iSMA_control::Drivers</ForComponentType>
</ForComponentTypes>
```

Figure 185. Creating custom table view

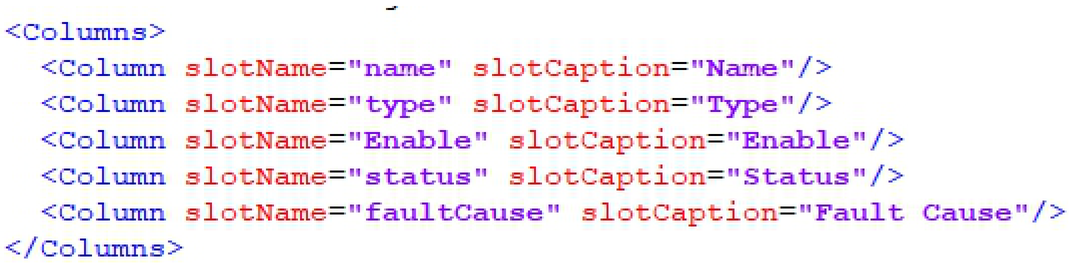
- <Filtercomponents>–defines the names of components, which will appear in the table on the defined view and limits the number of components shown in the Device Kits window;



```
<FilterComponents>
  <FilterComponent>iSMA_BACnet::BACnetNetwork</FilterComponent>
  <FilterComponent>iSMA_DALI::DALINetwork</FilterComponent>
  <FilterComponent>iSMA_localIO::localIO</FilterComponent>
  <FilterComponent>iSMA_LCD::Display</FilterComponent>
  <FilterComponent>iSMA_ModbusAsyncNetwork::ModbusAsyncNetwork</FilterComponent>
  <FilterComponent>iSMA_ModbusTcpNetwork::ModbusTcpNetwork</FilterComponent>
  <FilterComponent>iSMA_ModbusTcpSlaveNetwork::ModbusTcpSlaveNetwork</FilterComponent>
  <FilterComponent>iSMA_obix::Obix</FilterComponent>
  <FilterComponent>iSMA_OneWire::OneWireNetwork</FilterComponent>
  <FilterComponent>iSMA_p2p::P2pNetwork</FilterComponent>
  <FilterComponent>iSMA_MBus::MbusIpNetwork</FilterComponent>
  <FilterComponent>iSMA_MBus::MbusLocalNetwork</FilterComponent>
  <FilterComponent>iSMA_platFCU::SlaveNetwork</FilterComponent>
  <FilterComponent>iSMA_platFCU::localIO</FilterComponent>
  <FilterComponent>iSMA_BACnetMasterSlave::BACnetMasterSlaveNetwork</FilterComponent>
</FilterComponents>
```

Figure 186. Creating custom table view

- <Columns>–a group of settings allowing to read particular slots from components assigned to particular view along with the name replacing the name of the slot;



```
<Columns>
  <Column slotName="name" slotCaption="Name"/>
  <Column slotName="type" slotCaption="Type"/>
  <Column slotName="Enable" slotCaption="Enable"/>
  <Column slotName="status" slotCaption="Status"/>
  <Column slotName="faultCause" slotCaption="Fault Cause"/>
</Columns>
```

Figure 187. Creating custom table view

The following actions need to be taken to extend the offered tabular drivers views with custom networks, devices; and points:

- In 'drivers.xml' file extend the list <Filtercomponents> with component type defining new network (driver) unless such network has already been entered.
- Duplicate any existing '……DeviceManager.xml' file to define the devices view and optionally change the file name.
- Open newly created file, change the name of the view under the tag <ViewName> and properly edit the definition inside the tags: <ForcomponentTypes>, <Filtercomponents> and <Columns>.
- Duplicate any existing '……PointManager.xml' file to define the points view for a particular device and optionally change the file name .
- Open newly created file, change the name of the view under the tag <ViewName> and properly edit the definition inside the tags: <ForcomponentTypes>, <Filtercomponents> and <Columns>.

Close and restart the iSMA Tool, as the iSMA Tool checks drivers views definitions only on start-up.

Warning!

A thorough description of tags is also available in the drivers.xml file. Getting to know it will allow proper configuration of this and other .xml files.

Warning!

The iSMA Tool reads all .xml configuration files for tabular views only during start-up, and if an error is found in the .xml file syntax, it will display a proper information in the Console.

14 iSMA-B-AAC20 Simulator

Thanks to the iSMA Tool software, the iSMA-B-AAC20 device can be programmed in the offline mode using the iSMA-B-AAC20 simulator.

The latest simulator can be downloaded at https://ismacontrolli.com.

Windows compatibility – Simulator is an application intended for Windows operating system only, and can be run on Windows 7, 8, and 10 series.

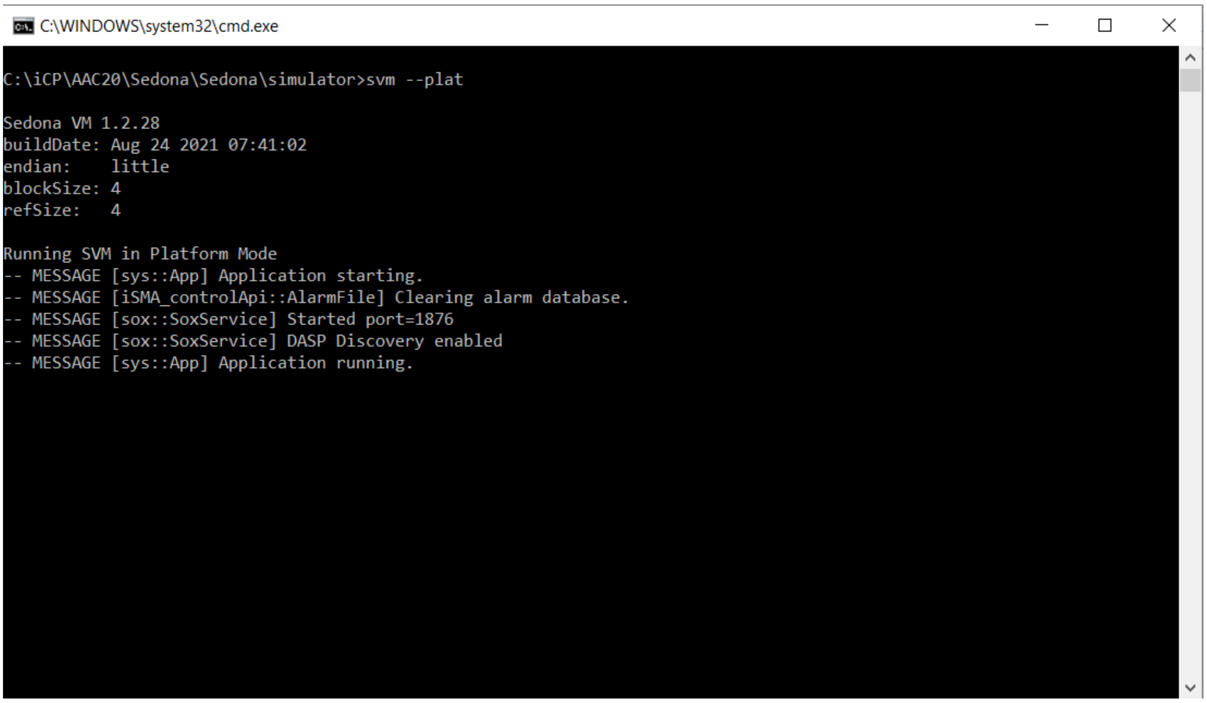
14.1 Running App on a Simulated Device

The Sedona Device Simulator tool lets to run a Sedona app on the iSMA Tool as a simulated device using the Sedona device vendor-supplied simulator, SVM (Sedona Virtual Machine).

This procedure describes how to run an app in the Sedona Device Simulator.

14.1.1 Start

- Run a "run" bat file from a Simulator folder to launch the simulator program:



```
C:\iCP\AAC20\Sedona\Sedona\simulator>svm --plat

Sedona VM 1.2.28
buildDate: Aug 24 2021 07:41:02
endian:    little
blockSize: 4
refSize:   4

Running SVM in Platform Mode
-- MESSAGE [sys::App] Application starting.
-- MESSAGE [iSMA_controlApi::AlarmFile] Clearing alarm database.
-- MESSAGE [sox::SoxService] Started port=1876
-- MESSAGE [sox::SoxService] DASP Discovery enabled
-- MESSAGE [sys::App] Application running.
```

Figure 188. Launching the simulator

- Open a direct Sox connection using a localhost in the iSMA Tool. Select Add Device.

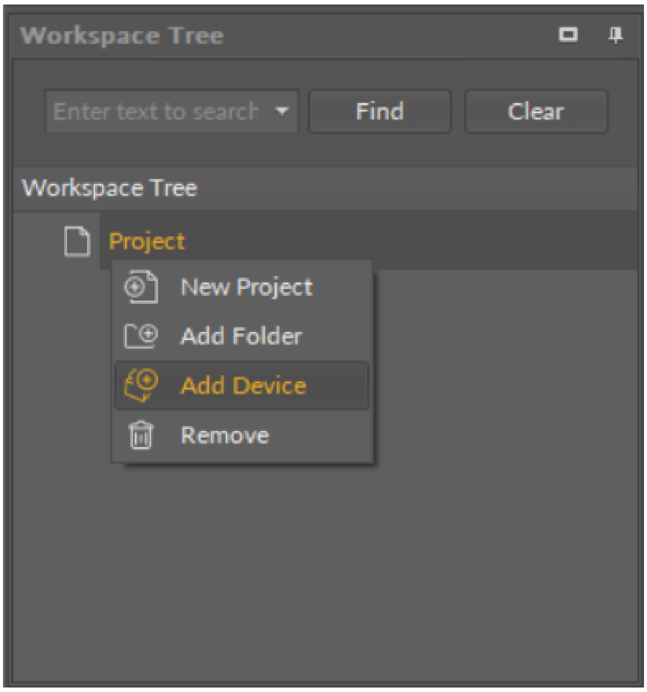
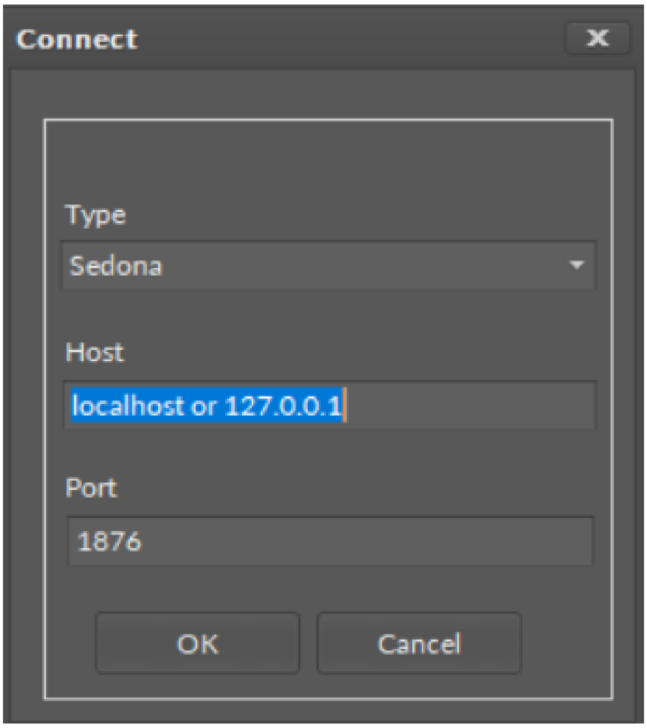


Figure 189. Add device

- In the Connect dialog box, for Host IP, enter: localhost. In the Authentication window, in the Username and Password fields, enter the credentials used in the new app file that has been created, and click Finish.



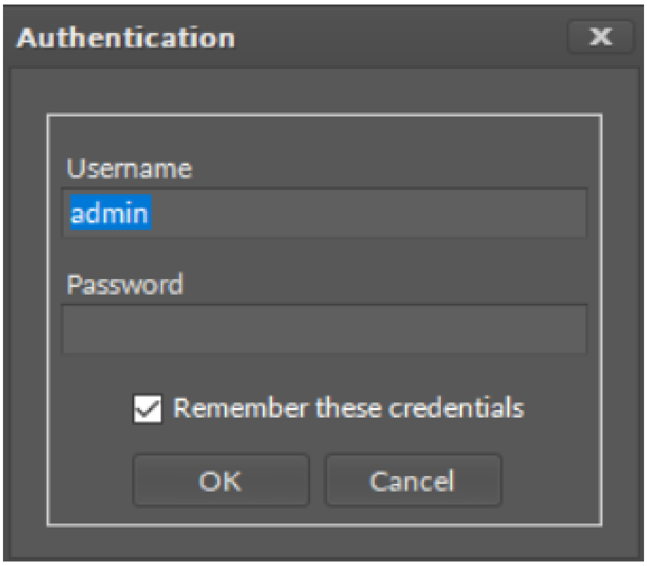
Type: Sedona

Host: localhost or 127.0.0.1

Port: 1876

Figure 190. The Connect dialog window

If the sample app is used on the sample simulator, the default credentials should be entered: Username: admin, Password: <blank>.



Authentication

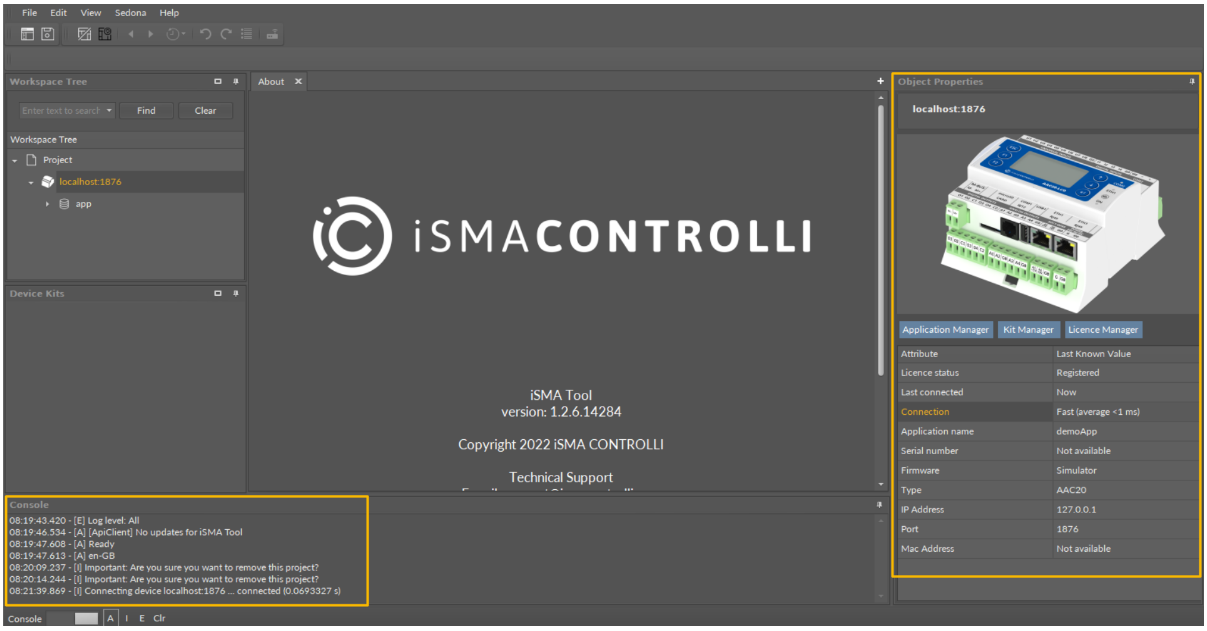
Username
admin

Password

☑ Remember these credentials

Figure 191. The Authentication dialog window

- Once connected, a direct Sox session is established with a connection to the app running in the simulated device:



iSMA Tool
version: 1.2.6.14284

Copyright 2022 iSMA CONTROLLI

Technical Support

localhost:1876

Attribute	Last Known Value
Licence status	Registered
Last connected	Now
Connection	Fast (average <1 ms)
Application name	demoApp
Serial number	Not available
Firmware	Simulator
Type	AAC20
IP Address	127.0.0.1
Port	1876
Mac Address	Not available

Console

```
08:19:43.420 - [E] Log level: All
08:19:46.534 - [A] [ApiClient] No updates for iSMA Tool
08:19:47.608 - [A] Ready
08:19:47.613 - [A] en-GB
08:20:09.237 - [I] Important: Are you sure you want to remove this project?
08:20:14.244 - [I] Important: Are you sure you want to remove this project?
08:21:39.869 - [I] Connecting device localhost:1876 ... connected (0.0693327 s)
```

Figure 192. Connected simulator

14.2 Making Changes to the App Running on a Simulated Device

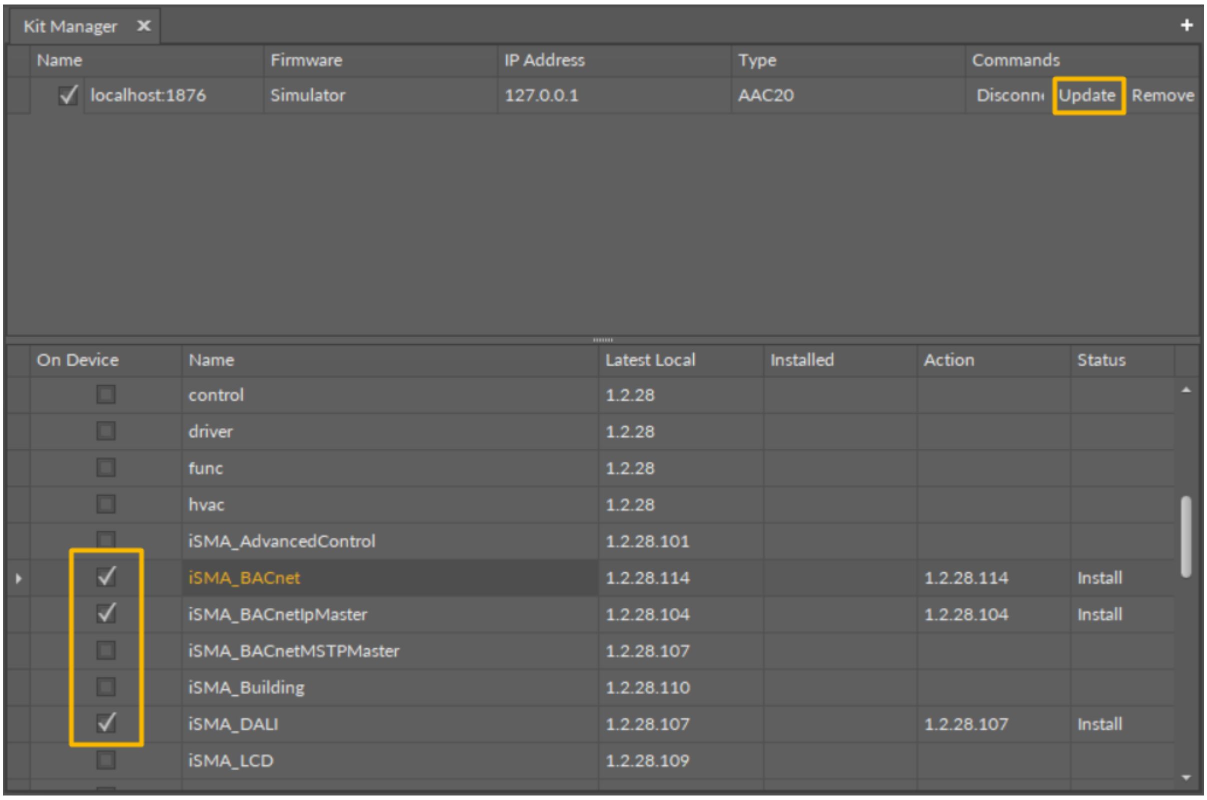
This procedure provides an example of making changes in the app running on a simulated device using the Kit Manager tool under the device to add a kit to the app.

14.2.1 Requirements

- The app must be successfully running in the device simulator.
- There is a running Sox connection to the app currently executed on the simulated device.

14.2.2 Steps

- In the Objects properties, click the Kit Manager to launch the tool.
- In the Kit Manager view, click the checkbox to select the kit, and click Update.



	Name	Firmware	IP Address	Type	Commands
✓	localhost:1876	Simulator	127.0.0.1	AAC20	Disconn Update Remove

On Device	Name	Latest Local	Installed	Action	Status
□	control	1.2.28			
□	driver	1.2.28			
□	func	1.2.28			
□	hvac	1.2.28			
□	iSMA_AdvancedControl	1.2.28.101			
✓	iSMA_BACnet	1.2.28.114		1.2.28.114	Install
✓	iSMA_BACnetIpMaster	1.2.28.104		1.2.28.104	Install
□	iSMA_BACnetMSTPMaster	1.2.28.107			
□	iSMA_Building	1.2.28.110			
✓	iSMA_DALI	1.2.28.107		1.2.28.107	Install
□	iSMA_LCD	1.2.28.109			

Figure 193. Installing new kits in the Kit Manager

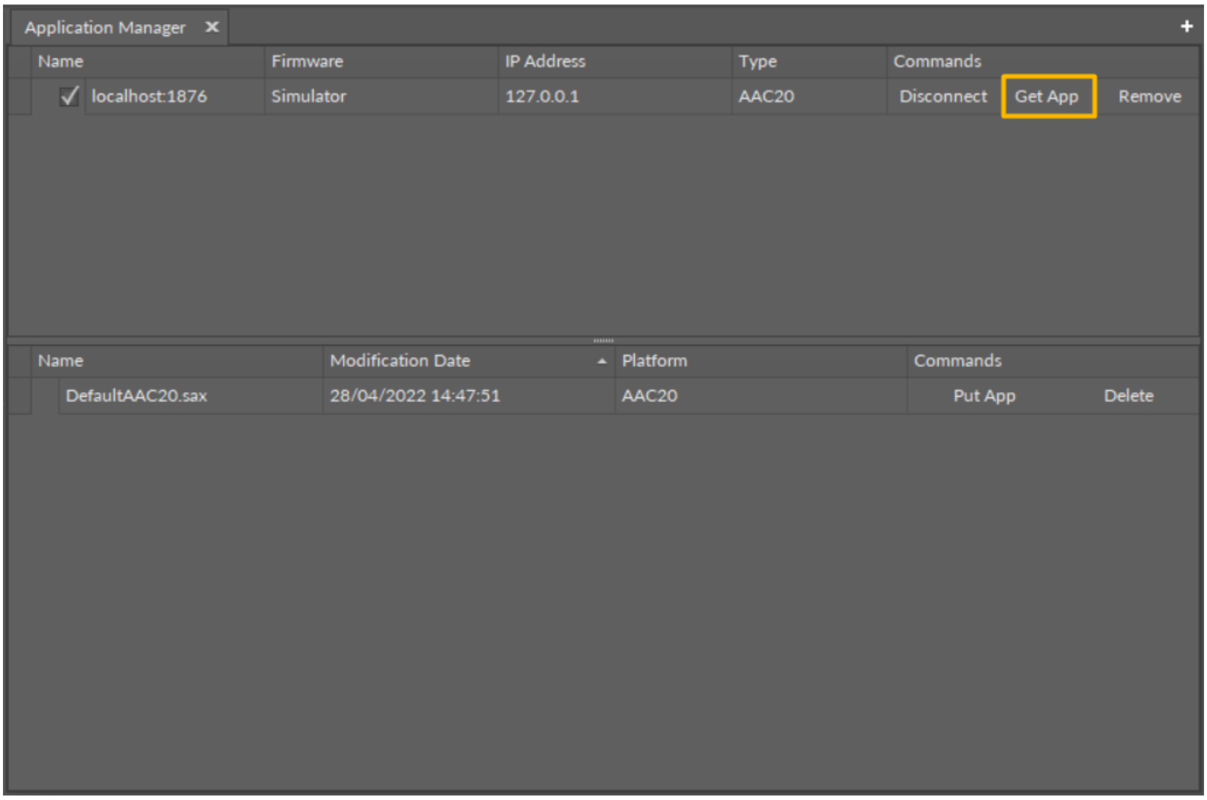
- When kit changes have been successfully completed, the simulator automatically restarts and connects.

14.3 Saving the Modified Simulator App File

This procedure describes how to stop the simulation and save the modified app.

14.3.1 Steps

- In the Objects Properties, click the Application Manager to launch the tool.
- In the Application Manager view, click the Get App option to save the application, and click Update.



Name	Firmware	IP Address	Type	Commands		
✓ localhost:1876	Simulator	127.0.0.1	AAC20	Disconnect	Get App	Remove

Name	Modification Date	Platform	Commands	
DefaultAAC20.sax	28/04/2022 14:47:51	AAC20	Put App	Delete

Figure 194. Saving application

- The modified app has been saved successfully to a selected location.

14.4 Scope of Simulator's Actions

14.4.1 Scope of Actions in Managers:

- Application Manager and License Manager: full functionality;
- Kit Manager: not supported (view mode only).

14.4.2 Scope of Actions in Services:

- Plat: all options functioning (working on localhost, some options are set to output values);
- Users: adding, removing, and editing of users functioning;
- Sox: all options are configurable and saved;
- Time: all options are configurable and saved;
- Alarm: all options are configurable and saved;
- History: all options are configurable and saved;
- Drivers: not available;
- Logic:
    - Sys: all options functioning;
    - iSMA_ControAPI: all options functioning;
    - iSMA_Control: scope of options is limited in regards to integer elements, which are set to a minimum value and cannot be configured;
    - iSMA_platAAC20: not implemented in the simulator;

15 iSMA-B-FCU in the iSMA Tool

This section describes how to control the iSMA-B-FCU device in the iSMA Tool.

15.1 Software Requirements

The iSMA Tool requires a connection with the iSMA-B-FCU using a dedicated software, the FCU Updater. The latest software can be downloaded here: https://ismacontrolli.com.

15.1.1 Windows Compatibility

The FCU Updater is an application intended for Windows operating system only, which can be run on Windows 7, 8, and 10 series.

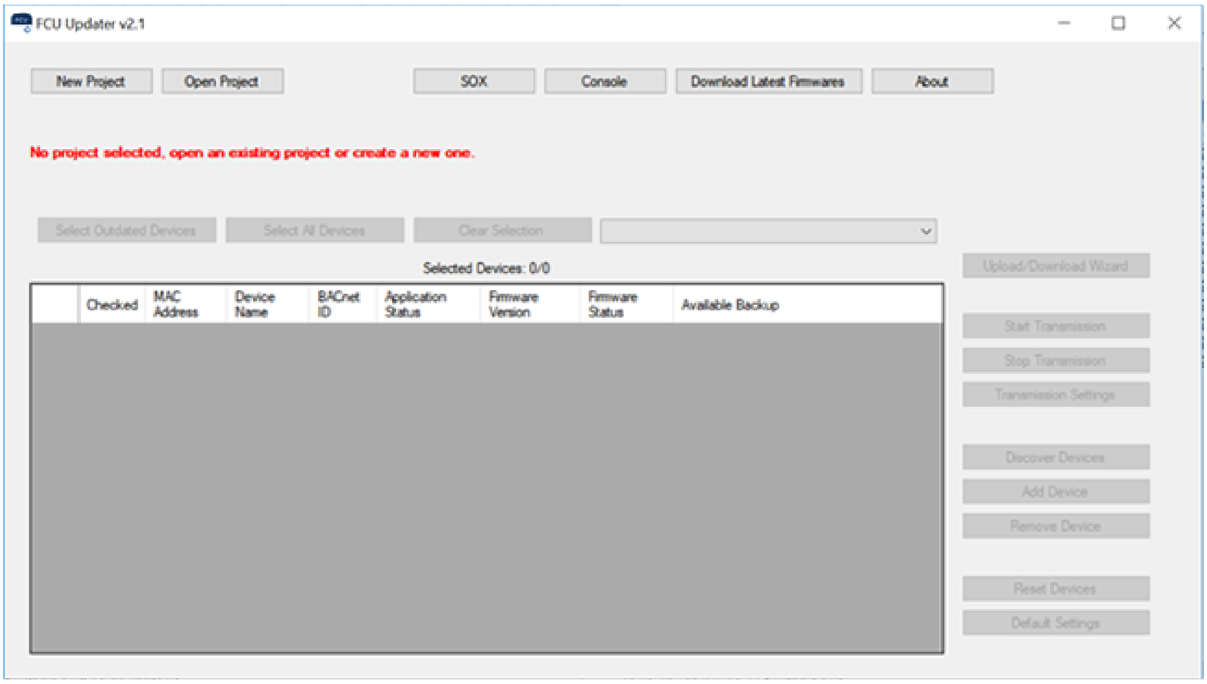
> **Warning!**
>
> To use all features of FCU Updater the latest Oracle Java 32-bit is required.

Please, download the latest Java software from Oracle Download website first.

15.2 SOX Protocol for Connecting iSMA-B-FCU with iSMA Tool

In order to establish a connection between the iSMA-B-FCU controller and iSMA Tool, the SOX connection needs to be opened.

The SOX function allows online programming of the iSMA-B-FCU controller (real time). The controller must be connected directly via a USB cable. The SOX button is always enabled each time the USB cable is connected directly to the iSMA-B-FCU controller and opening of any project is not required.



No project selected, open an existing project or create a new one.

Selected Devices: 0/0

	Checked	MAC Address	Device Name	BACnet ID	Application Status	Firmware Version	Firmware Status	Available Backup

Figure 195. Opening SOX protocol

The SOX button guides the user to the next window, where the SOX protocol can be started by pressing the 'Begin Communication' button. After starting the communication to the connected controller, the User can edit or build the application inside the iSMA-B-FCU controller by using the iSMA Tool software.

The SOX window has the following communication parameters:

- Port: port number (usually 1876) used by SOX protocol;
- Begin Communication button: start SOX communication. It is recommended to open SOX communication with the iSMA-B-FCU controller before running the iSMA Tool software;
- Stop Communication button: stop SOX communication.

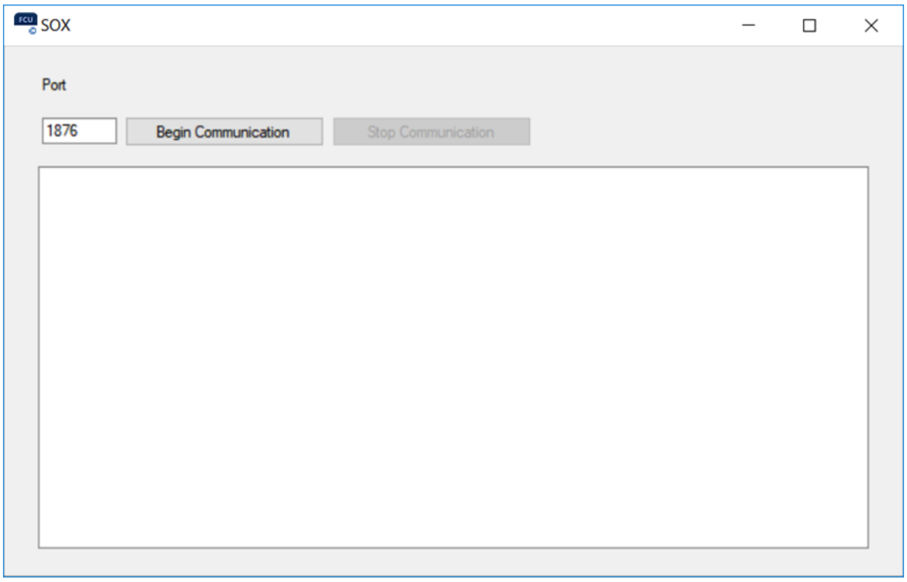
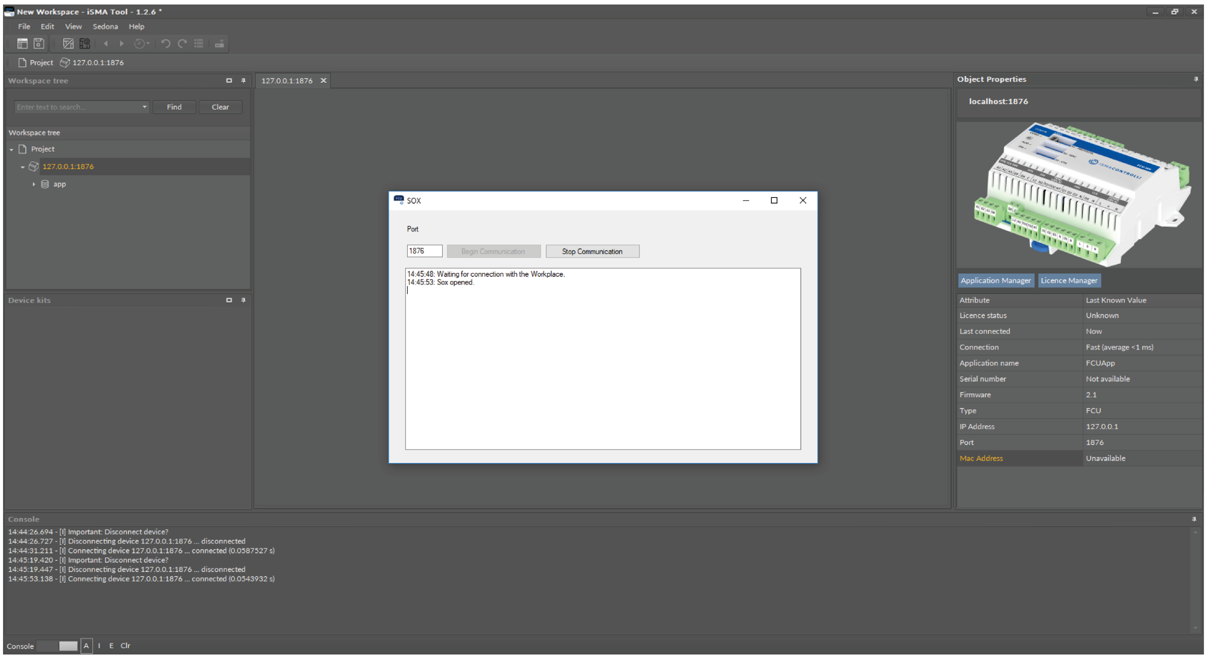


Figure 196. SOX protocol console

Note: Only one instance of the SOX protocol console can be opened at a time. If the SOX button in the main application window is disabled, the SOX protocol console has been already opened, and probably it is hidden somewhere under other windows on the user's screen.

If all the above actions have been performed, the communication via SOX protocol can be established with the iSMA-B-FCU controller using the iSMA Tool software.



Port: 1876

```
14:45:48: Waiting for connection with the Workplace.
14:45:53: Sox opened.
```

Attribute	Last Known Value
Licence status	Unknown
Last connected	Now
Connection	Fast (average <1 ms)
Application name	FCUApp
Serial number	Not available
Firmware	2.1
Type	FCU
IP Address	127.0.0.1
Port	1876
Mac Address	Unavailable

```
14:44:26.694 - [I] Important: Disconnect device?
14:44:26.727 - [I] Disconnecting device 127.0.0.1:1876 ... disconnected
14:44:31.211 - [I] Connecting device 127.0.0.1:1876 ... connected (0.0587527 s)
14:45:19.420 - [I] Important: Disconnect device?
14:45:19.447 - [I] Disconnecting device 127.0.0.1:1876 ... disconnected
14:45:53.138 - [I] Connecting device 127.0.0.1:1876 ... connected (0.0543932 s)
```

Figure 197. Open communication beetwen iSMA-B-FCU device and iSMA Tool

Note: A complete iSMA-B-FCU controller programming manual can be downloaded here: https://ismacontrolli.com.

16 Keyboard Shortcuts

Shortcut	Description
Ctrl + N	New Workspace (creates new workspace on Workspace Tree)
Ctrl + O	Opens a dialog window to choose a *.itw file saved earlier (Workspace)
Ctrl + S	Save changes in Workspace Tree window tree to a defined workspace in *.itw file. Does not apply to saving data in controller.
Ctrl + Q	Exiting iSMA Tool with confirmation of the intent to leave.
Ctrl + C	Copying of the selected Components.
Ctrl + V	Pasting of the Components copied with (Ctrl + C) option.
Ctrl + D	Duplication of selected Components. The option does not overwrite what was remembered during (Ctrl + C) copying.
Delete	Removal of selected elements (Component and/or Links).
Ctrl or Shift	The keys allow to select Components and Links selectively. Ctrl adds the chosen element, and Shift defines a set of added elements.
Ctrl + N	Create new Workspace Tree window for new projects
F	Rescale all Components on Wire Sheet view to see the all on the screen
Ctrl + mouse wheel	Zoom Wire Sheet in/out
Ctrl + A	Select all objects on active view
F11	Hide/show all other windows to make main screen as large as possible
F12	Restore previous windows position and size before used F11 option
Ctrl + Tab	Change focus to next window within opened windows in iSMA Tool
Ctrl + Shift + Tab	Change focus to previous window within opened windows in iSMA Tool
Tab	Navigate to the next object in current tree or list
Shift +Tab	Navigate to the previous object in current tree or list
PageUp/ PageDown	Jump to the first/last object in current tree or list

Ctrl + F	Use filter on active view
F2	Rename selected object
F10	Change focus to the main menu with a possibility to navigate between menu items
Ctrl + P	Print/export current view (option not available on all views)
Ctrl + Left/Right	Expand/collapse the tree
Esc	Cancel current settings
Enter	Accept current settings
Alt + F4	Exits iSMA Tool with prompt popup
Ctrl + Z	Undo previous action on components
Ctrl + Y	Redo previously undone action on components
Alt + Left/Right	Navigate view history
Alt + Shift + Left/Right	Show view history flyout panel

Table 3. Useful keyboard shortcuts