

Emalytics and ILC 2250 BI – getting started

Quick-start guide

Revision 01

“Quick-start guide”

“Emalytics and ILC 2250 BI – getting started”

UM QS EN EMALYTICS ILC 2250 BI, Revision 01 2026-06-10

This manual is valid for:

Designation	As of revision (HW)	As of revision (FW)	Item no.
ILC 2250 BI	01		1541303
ILC 2250 BI-L	01		1535543
Emalytics 1.4 and higher	XX	XXX	XXXXXXX

112333_en_01

Phoenix Contact GmbH & Co. KG • Flachsmarktstraße 8 • 32825 Blomberg • Germany
phoenixcontact.com

Table of contents

1 For your safety 9
1.1 Identification of warning notes 9
1.2 Qualification of users 9
1.3 Field of application of the product 10
1.3.1 Intended use 10
1.3.2 Product changes 10
1.4 Safety notes 10

2 Emalytics Automation Framework 13
2.1 Emalytics Automation host 13
2.2 Emalytics Automation Platform 13
2.3 Emalytics Automation modules 14
2.4 Emalytics Automation Station 14
2.5 Emalytics Automation system structure 15

3 Emalytics Automation Workbench 16
3.1 Emalytics Automation Supervisor installation 17
3.2 Installing Emalytics add-ons 17
3.3 Emalytics Automation Supervisor file system 18
3.4 Emalytics Automation Supervisor licensing 18
3.5 System passphrase 18
3.6 Starting the Workbench 19
3.7 Help 20

4 ILC 2250 BI 21
4.1 Hardware installation 21
4.1.1 Power supply 21
4.1.2 Network connection 21
4.1.3 USB-C port 21
4.2 Commissioning tools 22
4.2.1 Web-based management (WBM) 22

5 Commissioning of Emalytics Automation (tutorial) 23
5.1 Preparation 23
5.1.1 Version 23
5.1.2 License 23
5.1.3 New station 23
5.1.4 Establishing the platform connection 25

5.2 Parameterizing commissioning steps ... 29
5.2.1 Licensing ... 29
5.2.2 Runtime profiles ... 31
5.2.3 Installing a station ... 32
5.2.4 Install the modules ... 33
5.2.5 Installing the system software ... 34
5.2.6 Setting the time ... 34
5.2.7 IP settings ... 35
5.2.8 Configuring platform users ... 36
5.3 Completing commissioning ... 37

6 Station engineering (tutorial) ... 39
6.1 Components ... 39
6.2 Views ... 39
6.3 Station connection ... 40
6.4 Hello world – first application ... 42
6.4.1 Installing the module ... 42
6.4.2 Creating the component ... 43
6.4.3 Configuring the component ... 44
6.4.4 Connecting components – links ... 45
6.5 Points ... 46
6.5.1 Point value and status ... 46
6.5.2 Writable points ... 47
6.5.3 Read-only points ... 48
6.5.4 Facets ... 48
6.6 Trend data collection – history ... 49
6.7 Alarm management ... 53
6.8 Scheduling ... 56
6.9 Web visualization ... 57
6.9.1 User-defined views ... 57
6.9.2 User management ... 65
6.10 Drivers ... 69
6.10.1 I/O driver – SysmikScaIoNetwork ... 70
6.10.2 Serial drivers ... 73
6.11 Saving a station ... 74
6.11.1 Saving a station ... 74
6.11.2 Copying a station ... 75
6.11.3 Backup ... 75
6.12 Application structure ... 76

7 Internal communication within the Emalytics network ... 78
7.1 Emalytics Automation Supervisor Station ... 78

7.2 Communication ... 79
7.3 Integrating a station ... 80
7.3.1 Fox ... 80
7.3.2 Foxs ... 80
7.4 Integrating points into the Supervisor Station ... 82
7.5 Integrating trend data into the Supervisor Station ... 82
7.6 Integrating schedules into the Supervisor Station ... 83
7.7 Central alarms ... 83

8 Security ... 84
8.1 Passwords ... 84
8.2 Access data ... 84
8.3 Protocols ... 84
8.4 Passphrase ... 84
8.5 Environment ... 84

9 Diagnostics ... 85
9.1 Version information ... 85
9.2 Event logging ... 85
9.2.1 Emalytics Station ... 86
9.2.2 Emalytics Automation Workbench ... 86
9.3 Execution of applications ... 87
9.4 System diagnostics – Spy ... 88
9.4.1 EngineManager ... 88
9.4.2 ClassLoaders ... 88
9.4.3 Log setup ... 88

10 Software maintenance ... 89
10.1 Emalytics Automation Workbench ... 89
10.2 Emalytics Automation Supervisor ... 89
10.3 IB IL 2050 BI Emalytics Controller ... 89

A Technical appendix ... 91

B Appendixes ... 93
B 1 List of figures ... 93
B 2 List of tables ... 97
B 3 Index ... 99

1 For your safety

Read this manual carefully and keep it for future reference.

1.1 Identification of warning notes



This symbol indicates hazards that could lead to personal injury.

There are three signal words indicating the severity of a potential injury.

DANGER
Indicates a hazard with a high risk level. If this hazardous situation is not avoided, it will result in death or serious injury.

WARNING
Indicates a hazard with a medium risk level. If this hazardous situation is not avoided, it could result in death or serious injury.

CAUTION
Indicates a hazard with a low risk level. If this hazardous situation is not avoided, it could result in minor or moderate injury.



This symbol together with the **NOTE** signal word warns the reader of actions that might cause property damage or a malfunction.



The symbol informs you that you have to observe the instructions. Only install and operate the device once you have familiarized yourself with its properties by means of the user documentation.



Here you will find additional information or detailed sources of information.

1.2 Qualification of users

The use of products described in this manual is oriented exclusively to:

- Electrically skilled persons or persons instructed by them. The users must be familiar with the relevant safety concepts of automation technology as well as applicable standards and other regulations.
- Qualified application programmers and software engineers. The users must be familiar with the relevant safety concepts of automation technology as well as applicable standards and other regulations.

1.3 Field of application of the product

1.3.1 Intended use

The ILC 2250 BI controller is a modular small-scale controller that can be used for smaller and medium-sized applications. The device has IP20 protection class and is designed for use in closed control cabinets or control boxes (junction boxes) with IP54 degree of protection or higher.
The device is designed for use in industrial environments.

1.3.2 Product changes

Modifications to hardware and firmware of the device are not permitted.

Incorrect operation or modifications to the device can endanger your safety or damage the device. Do not repair the device yourself. If the device is defective, please contact Phoenix Contact.

1.4 Safety notes

Observe the country-specific installation, safety, and accident prevention regulations.

During startup and maintenance work, proceed in accordance with the five safety rules of DIN EN 50110-1. In general, the rules should be observed in the specified order:

– Disconnect from power source
– Secure against being switched on again
– Determine absence of voltage
– Ground and short-circuit
– Cover or safeguard adjacent live parts

Once the work is complete, perform the above steps again in reverse order.



NOTE: Property damage due to incorrect use

The IP20 degree of protection (IEC 60529/EN 60529) requires that the device is used in a clean and dry environment. Using the device in an environment that is outside of the specified limits may cause damage to the device.

Do not subject the device to mechanical and/or thermal stress that exceeds the specified limits.

NOTE: Risk of unauthorized network access

Connecting devices to a network via Ethernet always entails the risk of unauthorized access to the network.

Therefore, please check for the option of disabling active communication channels in your application (for instance SNMP, FTP, BootP, DCP, HTTP, HTTPS, etc.) or setting passwords to prevent third parties from accessing the controller without authorization and modifying the system.

Due to its communication interfaces, the controller should not be used in safety-critical applications without additional security appliances.
Please take additional precautions in accordance with the IT security requirements and standards applicable to your application (e.g., virtual networks (VPN) for remote maintenance access, firewalls, etc.) to prevent unauthorized network access.

On first request, you shall release Phoenix Contact and the companies associated with Phoenix Contact GmbH & Co. KG, Flachsmarktstrasse 8, 32825 Blomberg, Germany in accordance with §§ 15 ff AktG (German Stock Corporation Act), hereinafter collectively referred to as “Phoenix Contact”, from all third-party claims made due to improper use.

For the protection of networks for remote maintenance via VPN, Phoenix Contact offers the mGuard product range of security appliances, a description of which is provided in the latest Phoenix Contact catalog (phoenixcontact.com/products).

Additional measures for protection against unauthorized network access are listed in the AH EN INDUSTRIAL SECURITY application note. The application note can be downloaded at phoenixcontact.com/products.

NOTE: Electrostatic discharge

Electrostatic discharge can damage or destroy components. When handling the device, observe the necessary safety precautions against electrostatic discharge (ESD) in accordance with EN 61340-5-1 and IEC 61340-5-1.

2 Emalytics Automation Framework

The **Emalytics Automation Framework** is a universal software kit for automation systems. As such, it makes up the central application of the **Emalytics building management system (BMS).**

It has a highly modular structure and provides ready-to-use functions for many standard requirements. The functionality can be extended in almost every imaginable direction by programming additional modules in Java.

Standard functions include:

- An engineering tool (**Emalytics Automation Workbench**) for all levels of the system
- Integration of information from lower-level third-party systems via numerous communication drivers
- Transfer of information to higher-level third-party systems via numerous communication drivers
- Data processing
- Alarm management
- Scheduling
- Trend data collection
- Process visualization
- Reports
- Data analysis
- Email/text messaging

Typical software extensions include special communication drivers as well as industry-specific modules and customizations from various manufacturers in the Niagara context (**Niagara** is a component of the Emalytics Automation Framework). The Niagara Community www.niagara-community.com is comprised of hardware and software manufacturers as well as users, system integrators, and training centers. This community offers products and services for the Niagara Framework.

2.1 Emalytics Automation host

The Emalytics Automation Framework runs on different hardware – called **hosts** in the Emalytics environment. There are different **Emalytics Controllers**, such as the **ILC 2250 BI**, which are already delivered with the Emalytics Automation Framework. There is also the **Emalytics Automation Supervisor**, which can be installed on computers with Windows or Linux operating systems, from simple PCs to virtual machines in the cloud.

2.2 Emalytics Automation Platform

A hardware- and OS-specific software application provides the **Emalytics Automation Platform** on every host. The platform is basically an operating system extension that provides a platform-independent Emalytics Automation environment. Using the platform, hosts can be centrally managed in the Emalytics Automation Workbench (or simply Workbench), the engineering tool of the Emalytics Automation Framework.

This includes:

– IP settings
– Software updates
– Installing, starting, stopping applications, creating backups
– License and certificate management

If an Emalytics Automation Supervisor for Windows operating systems is used, the platform consists of a Windows service called *Niagara*. It is also called the *platform daemon*.

2.3 Emalytics Automation modules

The Emalytics Automation Framework has a modular structure. For distinct groups of functions, software is provided in the form of modules. These Emalytics modules contain components that implement a specific function. They are very similar to libraries and corresponding function blocks in PLC programming.

Examples of modules include communication drivers that implement a communication protocol.

2.4 Emalytics Automation Station

In the engineering process, the Emalytics Automation Workbench is used to assemble an application from numerous components of various modules. This application is called Emalytics Automation Station (or simply station).

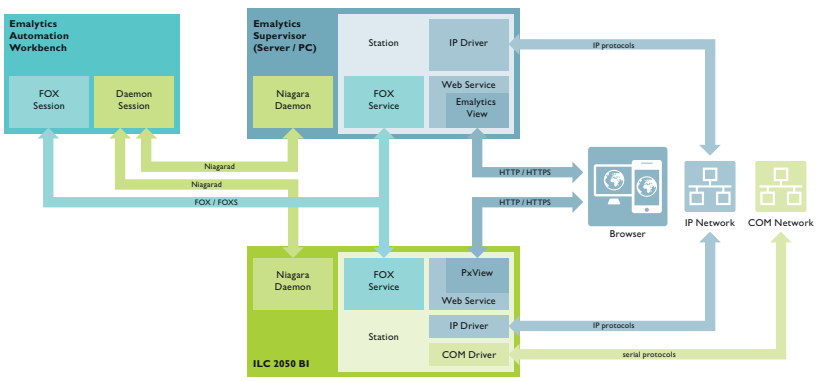


Figure 2-1 Station engineering in Emalytics

2.5 Emalytics Automation system structure

Emalytics Automation Stations communicate with each other. In typical system structures, Emalytics Controllers (e.g., ILC 2250 BI) perform a variety of functions in the field as integration platforms:

– Record and normalize field level data via serial or IP communication drivers
– Run local control algorithms
– Generate alarms
– Run schedules
– Record trend data

Several of these embedded stations may be connected to a central Supervisor station, which performs the following tasks:

– Integrate data from Emalytics Controllers
– Access smart devices via IP communication drivers
– Run central control algorithms
– Provide web-based process visualization
– Consolidate all alarms in an alarm console, alarm management
– Synchronize schedules of Emalytics Controllers
– Archive trend data of all Emalytics Controllers and export them to external databases, if necessary
– Central user management for the entire Emalytics system
– Central management of software updates
– Regular backups of all Emalytics Stations

Depending on the size of the automation system, additional hierarchy levels (buildings – property – country – continent – world) can be mapped in an Emalytics system.

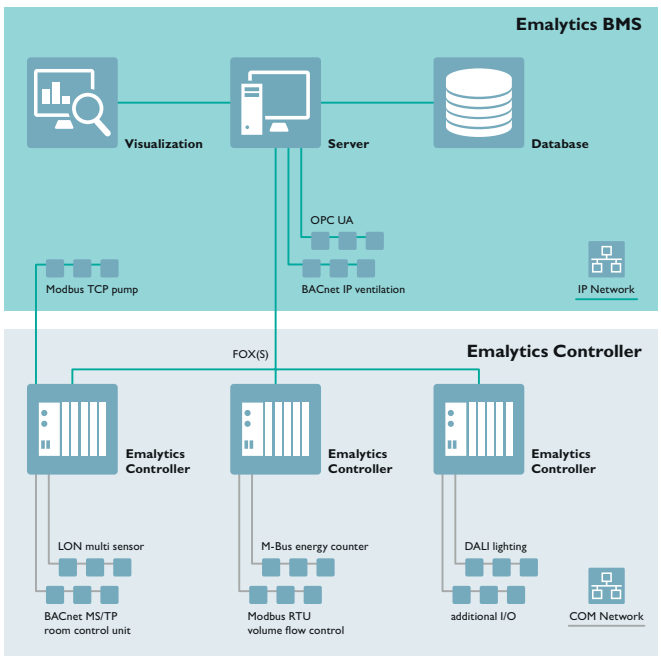


Figure 2-2 Example of Emalytics system structure

3 Emalytics Automation Workbench

The Emalytics Automation Workbench (or simply Workbench) is the central engineering environment in the Emalytics building management system (BMS). You can use the Workbench to manage all hosts via the platform and to program stations (applications).

The Workbench is not available as a separate product. Instead, it is part of every Emalytics Supervisor, installed with the Emalytics setup, and licensed with the Supervisor license.

The following system requirements apply for Emalytics Automation Supervisor installations (Niagara 4.x)

Depending on the application, number of integrated data points, data polling rate, number of simultaneous users, performance expectations, etc., supervisor installations can also run on less powerful platforms or may also need more powerful platforms.



Platform requirements:

- **Processor:** Intel® Xeon® 4th Gen. (or better), Intel® Core® i-series 12th Gen. (or better), AMD® Ryzen® 5 Gen. 3 (or better)
- **Memory:** At least 16 GB DDR4, 32 GB or more is recommended for larger systems
- **Hard drive:** At least 512 GB SSD, more is recommended depending on the specific archiving requirements
- **Network:** Ethernet adapter (10/100/1000 Mbps with RJ45 jack)
- **Video/display:** Minimum resolution 1080 p (1920 x 1080)
- **Operating system:** Windows 11 x64-bit; Windows Server 2025; Linux x64-bit: RedHat Enterprise Linux 9.x, Ubuntu 24.04 LTS
- **Browser:** Chrome, Firefox, Microsoft Edge
- **Mobile browser:** Safari on iOS, Chrome on Android
- **Relational database (optional):** Microsoft SQL Server 2022, MySQL Server 9.0
- **Container engine (for containerized supervisors)**: Docker, Inc.

The platform requirements for older Supervisor versions are included in the version notes for the respective version.

3.1 Emalytics Automation Supervisor installation

As well as the Emalytics add-ons, the Supervisor is installed as a component with the Emalytics setup.



The installation package for the Emalytics Supervisor can be downloaded as a .zip file under “Downloads” on the product page of the ILC 2250 BI controller at www.phoenixcontact.com. The Installation Wizard opens when you run the **Start.exe** file. Once you have confirmed the mandatory end user license agreement, you can select an installation directory
(default: C:\PhoenixContact\Emalytics<x.y>\... indicates the Emalytics version).
In the subsequent dialog boxes, you can select and launch the version you want to install (32-bit or 64-bit version).

Multiple versions of the Supervisor (including Workbench) can be installed on one computer. Whereas only one platform daemon (Windows service) can be active, all installed versions of the Emalytics Automation Workbench can run concurrently.
Newer versions of the Supervisor are usually always installed in dedicated installation folders. This allows you to continue using older versions without restrictions.



NOTE: Emalytics version station vs. Workbench

Always ensure that a station is only ever programmed with a Workbench of the same version. If this principle is not observed, there is a risk that your system may not function correctly.

3.2 Installing Emalytics add-ons

If you want to use the Workbench of an installed Supervisor for engineering Emalytics ILC 2250 BI Controllers, you must also install Emalytics add-ons in the installation directory of this Supervisor. Besides the Emalytics View visualization software, the Emalytics add-ons include Emalytics modules required to run the Emalytics Controllers as well as up-to-date firmware for the ILC 2250 BI controllers.

The following components are contained in the Emalytics add-ons:

- Emalytics View
- Modules required to run the Emalytics Controllers as well as up-to-date firmware for these controllers
- Additional application and driver modules for Emalytics Automation and Emalytics View

3.3 Emalytics Automation Supervisor file system

Every Supervisor installation creates a **home directory** (e.g., C:\PhoenixContact\Emalytics1.1\SupervisorN4-4.7.110.32). Program files of the Emalytics Platform and Workbench as well as Emalytics modules, including documentation, are stored here.

A **“Workbench User Home”** link refers to the home directory of the Workbench user, for example C:\Users\<User_Name>\Niagara4.7\PhoenixContact, where the Workbench user data is stored – especially stations, backups, and templates.

A **“Daemon User Home”** link refers to the home directory of the platform, for example C:\ProgramData\Niagara4.7\PhoenixContact, where especially the Supervisor or Emalytics Stations are stored.

3.4 Emalytics Automation Supervisor licensing

The Emalytics Automation Workbench can be started once the Emalytics installation is completed. If the Supervisor is not yet licensed, a message is displayed with the currently calculated host ID. You can now purchase a Supervisor license via the sales channel by specifying this host ID. Once generated, this license will be available on a license server and will be retrieved the next time you start the Workbench when connected to the Internet. Alternatively, you can obtain “*.license” files via email and copy them to the installation directory of the Supervisor, <Home>\security\licenses (typical example: “C:\PhoenixContact\Emalytics1.1\SupervisorN4-4.7.110.32\security\licenses”). You can then start and begin using the Workbench.

There are multiple Supervisor licenses that differ in the number of stations that can be integrated into the Supervisor. All of these licenses allow unrestricted use of the Workbench.

3.5 System passphrase

As a user, when you launch a Workbench for the first time on a machine, you are prompted to provide a system passphrase. This password is used to encrypt sensitive data of a station (e.g., password hashes of station users) in the **Workbench User Home** directory of the Workbench.

This system passphrase is important and should not be forgotten!

The hosts also have a system passphrase for encryption of sensitive data. Whenever a station is copied between the Workbench and host, the sensitive data is decrypted with the system passphrase of the source and immediately re-encrypted with the passphrase of the destination. If a passphrase is lost, the affected stations cannot be unlocked without losing sensitive data.

3.6 Starting the Workbench

The Emalytics Automation Workbench can be started via the start menu with or without the console. If certain errors occur, for example if the Workbench does not start, the console provides helpful diagnostic information. However, it is not needed for everyday use of the Workbench.

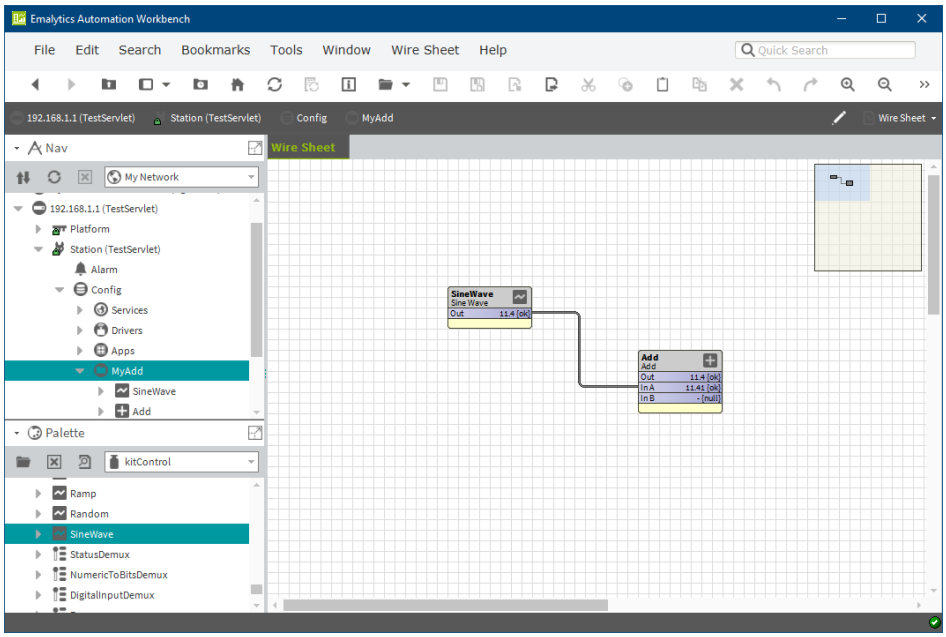


Figure 3-1 Workbench start screen

The Workbench window is divided into a central workspace and a side bar on the left. You can select the tools displayed in the side bar via the **“Window – Side Bars”** menu. **“Nav”** (navigator) is an important side bar. It contains a tree view of the Emalytics system, starting with the hosts, the various connections to them, and the available elements (tools, files, components). It enables easy navigation: When you double-click an element in the navigation tree, the default view of this element is displayed in the workspace. Other important options are provided via the local element-specific menu.

If necessary, you can open multiple sessions of the Workbench, for example by choosing **“File – New Window”** from the menu. Alternatively, you can create several independent workspaces using tabs by choosing **“File – New Tab”** from the menu. Actions in the left-hand navigation area are executed in a new Workbench window if you hold down the “Ctrl” key when double-clicking.

3.7 Help

The Emalytics Automation Supervisor includes a comprehensive help system. You can view the help system directly in the Workbench via the **“Window** – **Side Bars** – **Help”** menu. When you open the help for the first time, you must choose **“Load Help”** to load the help from the installed Emalytics modules.

You can display the help system in a new **“Niagara Help”** window by choosing **“Help – Help Contents”** from the menu. From the help menu, you can also jump to specific help topics for components or views currently displayed in the Workbench.

The help system has a modular structure. All available help modules are listed in the **“Table Of Contents”** tab. A good place to start is **“Doc User”**, which is essentially a manual.



Alternatively, all help is also available as PDF files in the installation folder of the Supervisor under **\docs**.

Further help can be found online on the Niagara Community website at www.niagara-community.com. Members of the community discuss topics related to the Framework in various forums on the website. Become a member of the community and post your own questions.

4 ILC 2250 BI

The ILC 2250 BI is a powerful modular controller with a wide variety of alignable Inline system terminals, including analog and digital inputs and outputs, interfaces for DALI, M-Bus, MP-Bus, meters, serial interfaces, etc.

For more detailed information, refer to the manual for the controller.

4.1 Hardware installation

The ILC 2250 BI is designed for snap-in mounting on DIN rails. At minimum, a power supply and Ethernet network are required for commissioning the ILC 2250 BI. In addition, a USB connection between the controller and PC can support the commissioning process.

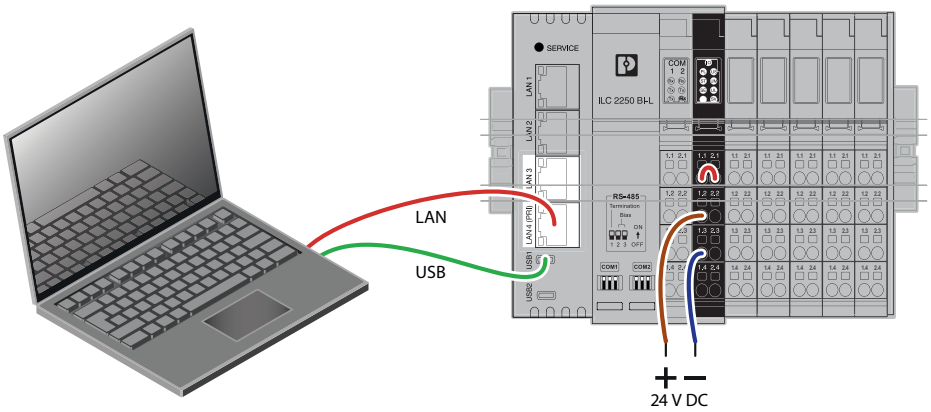


Figure 4-1 Commissioning the hardware

4.1.1 Power supply

The ILC 2250 BI is operated at 24 V DC. When connecting the controller, it is important to insert the additional wire bridge between contacts 1.1 and 1.2 of the supply terminal.

4.1.2 Network connection

You can configure the IP addresses of the controller and the assignment of the four Ethernet ports to the two internal Ethernet adapters of the controller. In the delivery state, the ILC 2250 BI can be reached via the primary eth0 Ethernet adapter on the LAN3 and LAN4 Ethernet ports at the default IP address 192.168.1.1.

We recommend using LAN4 for making initial contact with a controller.

4.1.3 USB-C port

USB1 is OTG-capable and therefore enables local IP communication for access to web-based management and the Niagara Workbench. To do this, connect the USB1 port of the controller to a USB port on your PC. This automatically installs a network adapter under Windows 11 (USB NCM host device).

4.2 Commissioning tools

4.2.1 Web-based management (WBM)

If a USB connection has been established between the ILC 2250 BI and the PC, you can open the commissioning website in a browser by entering **http://169.254.252.10:8443**. The commissioning website allows you to change general settings such as IP addresses and system time.

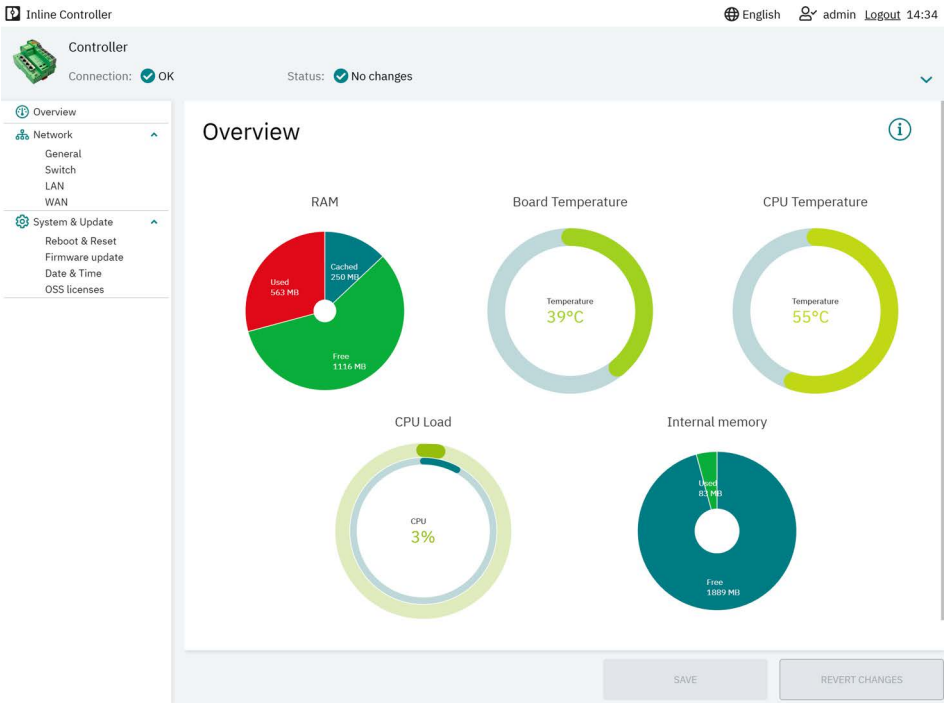


Figure 4-2 ILC 2250 BI web-based management home page

5 Commissioning of Emalytics Automation (tutorial)

Commissioning is the process of implementing the software for an Emalytics host. It includes licensing, installing the required software, and making various system settings. The following sections provide a tutorial that will familiarize you with the application.

5.1 Preparation

Commissioning involves several steps, some of which require preparation.

5.1.1 Version

First, the latest version of the Emalytics Automation Workbench (Section 3) as well as the Emalytics add-ons (Section 3.2) must be installed. This ensures that the latest software is installed on the host during commissioning. If you want to install a specific Emalytics version on the host, the appropriate Workbench version should be used. This may become necessary if software maintenance has expired and the Emalytics host does not have a license for the current Emalytics version.

5.1.2 License

To license the host, you must purchase a corresponding license via the sales channel by specifying the host ID of the Emalytics Automation host.

5.1.3 New station

In order to load a station during commissioning, you must first create it using the Workbench. Open the **“New Station Wizard”** by choosing **“Tools – New Station”** from the menu. Here, you must specify the station name under which the station will be stored in the “Workbench User Home” directory.

The station name becomes more important when multiple stations must communicate with each other via the Emalytics network. For this reason, station names must be unique throughout the network. However, the station name can be modified when stations are transferred from the Workbench to the host. In this way, a Workbench station can be copied to many hosts.

You can create stations using the Workbench or copy them from or to a host using the **Station Copier** platform tool.

Stations can be transferred via the file system, but this method is prone to errors because version checks are skipped and sensitive data is not encrypted and decrypted by means of a passphrase.

You must also select a station template. The **NewIlc2x50BiStation.ntpl** station template is suitable for the ILC 2250 BI.

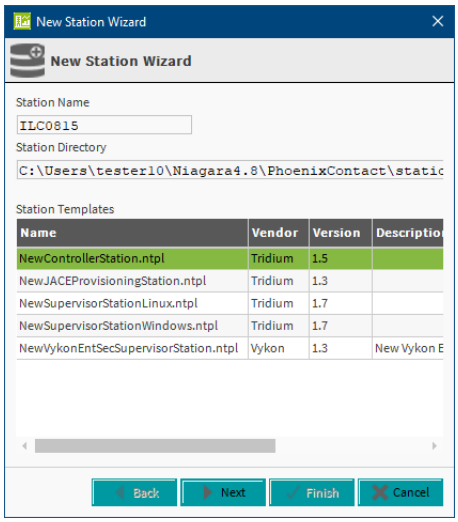


Figure 5-1 Starting the New Station Wizard

Through the high security standards of Emalytics, default passwords are avoided. When creating a station, the user is therefore requested to assign a password for the user **admin**. Observe the notes on page 26.

The **New Station Wizard** always assigns a user with the name **admin** to the station. A password must be entered and confirmed for this user. Please use the password **Admin12345** for the following tutorial. These login credentials will be used later for station engineering.

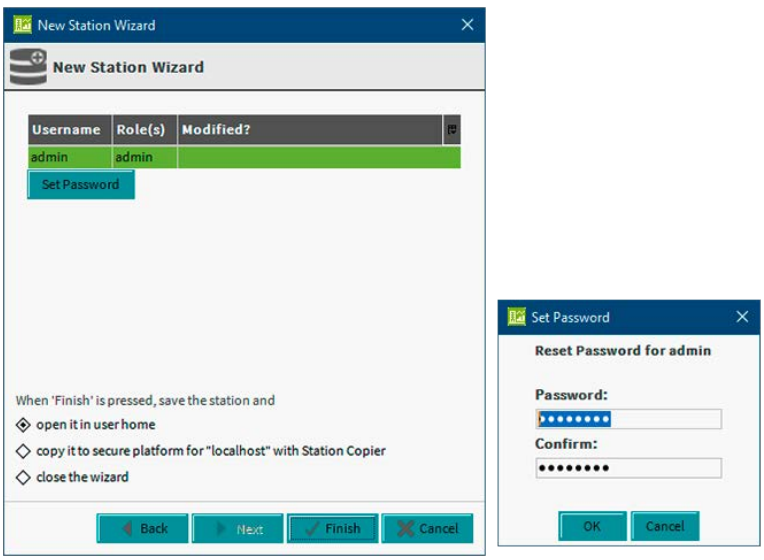


Figure 5-2 New Station Wizard administration

After you exit the New Station Wizard, the station is created in the Workbench User Home directory under “Stations” and displayed in the Workbench.

5.1.4 Establishing the platform connection

The connection to the platform can be established with TLS encryption or without encryption. Select “Platform Connection” for an unencrypted connection or “Platform SSL Connection” for an encrypted connection. The respective default port is automatically adapted.

We recommend SSL encryption especially for connections via the Internet or cloud systems.

For further information on security, please read the document “docStationSecurity”, which is part of the Emalytics installation.

Commissioning is done using the Emalytics Automation Workbench. For this, a connection is established to the platform of the host by choosing **“File – Open Platform”** from the menu.

- Select **“Platform Connection”** as the **session type**,
- enter the host IP address, and
- confirm your entry with **“OK”**.

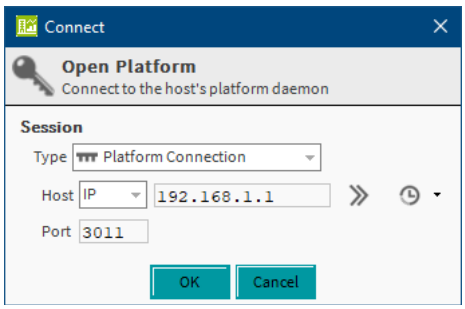


Figure 5-3 Platform connection establishment

During initial access to the platform via a secure connection, a self-signed standard certificate, which is to be confirmed after checking, is displayed. The small lock on the platform icon indicates a secure encrypted connection (see Figure 5-4 ff.).

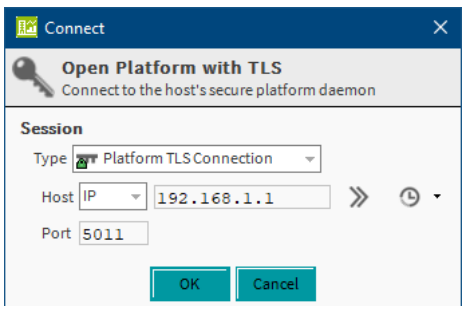


Figure 5-4 Platform SSL connection establishment

- You must now enter the user name and password for the platform connection. In the delivery state, the **user name** for the ILC 2250 BI is **phoenix** and the **password** is **inline**. Use your Windows login credentials for the Supervisor.

As of version 4.4, you will be requested to change the passphrase for the controller and the platform access data when first making a platform connection to an ILC 2250 BI.

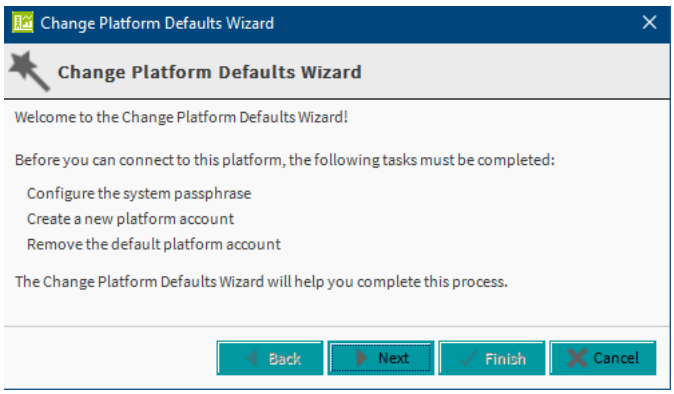


Figure 5-5 Change Platform Defaults Wizard

This system passphrase is used to encrypt sensitive data such as the password hashes of station users.

Enter an individual and secret passphrase that replaces the default passphrase, “in-line”.

The passphrase is subject to the same requirements as a platform password: It must be a minimum length of 10 characters and contain at least one number, one uppercase letter, and one lowercase letter.

A section in the document “docStation Security” is devoted to security. This document is part of the installation and can be called up via the online help.

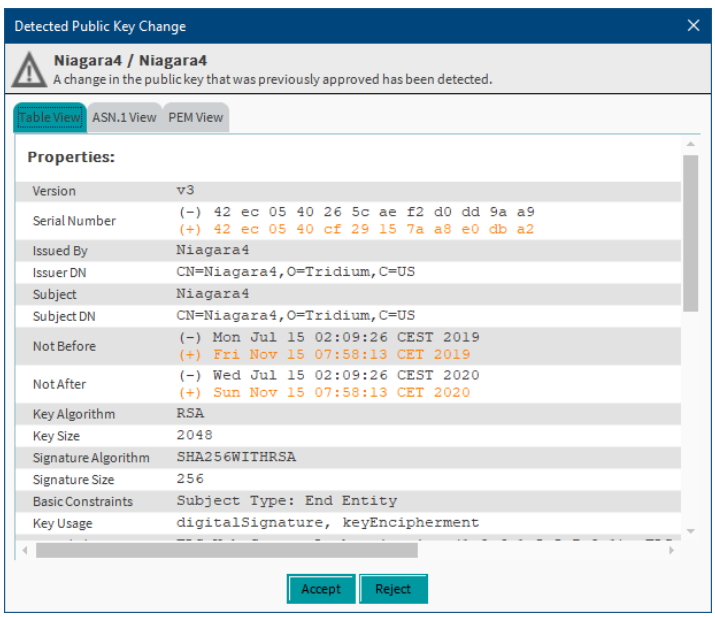


Figure 5-6 Certification confirmation

NOTE

This system passphrase is important and should not be forgotten!

You also require the current passphrase in order to change the passphrase later. **If you forget the passphrase, it can only be reset to “inline” by installing a clean distribution.**

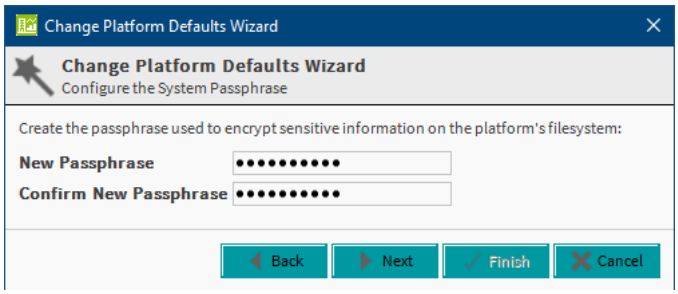


Figure 5-7 Parameterizing a passphrase

- Now define an individual and secret platform user with user name and password that replaces the default platform user, “phoenix”.

NOTE

The platform access data is important and must not be forgotten!

Store your access details securely. There is no way to reset the password without resetting the device to default settings.

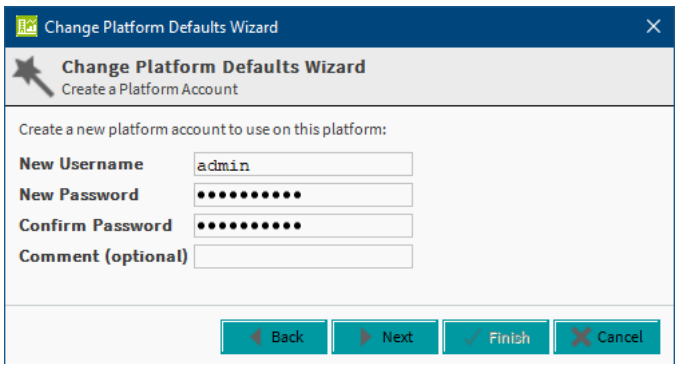


Figure 5-8 Parameterizing platform access data

All the parameterized changes are displayed again.

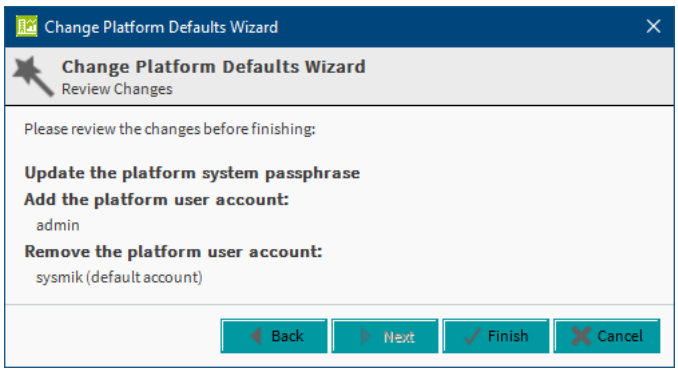


Figure 5-9 Summary of the changes

Confirm with “Finish” to execute the changes and use the platform connection.

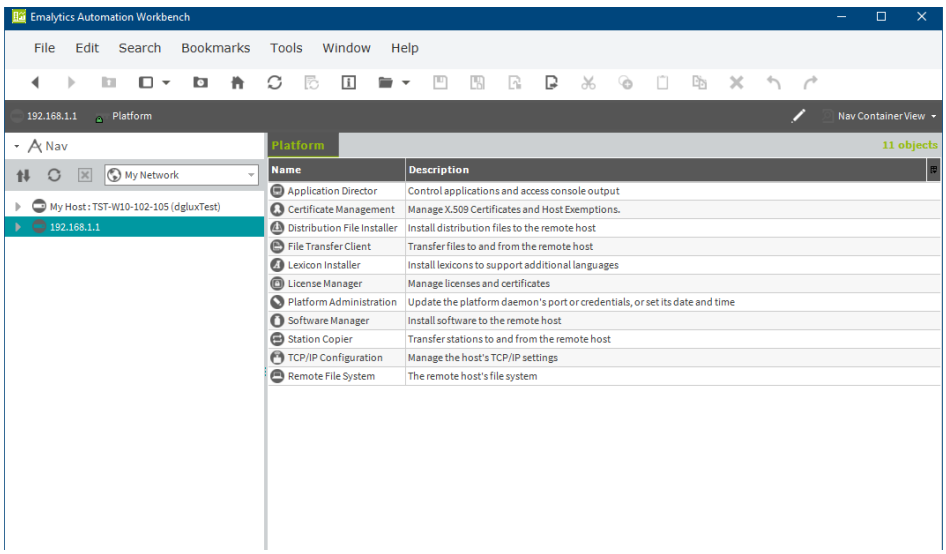


Figure 5-10 Commissioning: Platform tools

The list of platform tools is now displayed in the workspace of the Workbench.

5.2 Parameterizing commissioning steps

Commissioning involves many steps, all of which can be performed individually with the appropriate platform tools as needed.

The Commissioning Wizard, which you open by choosing **“Platform Administration – Commissioning”**, combines all of these steps into one operation.

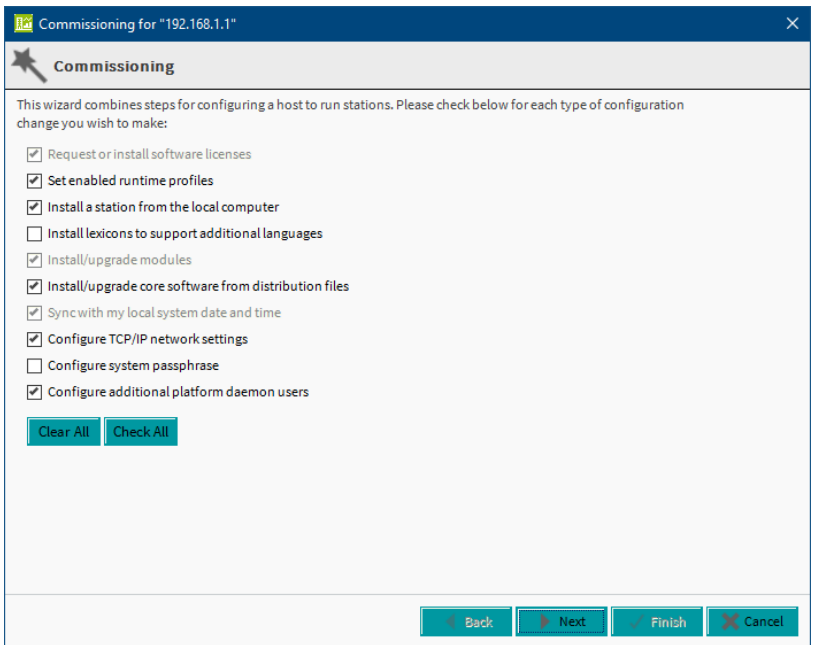


Figure 5-11 Commissioning checklist

Some steps are mandatory. Other steps can be manually enabled or disabled. You subsequently parameterize each enabled step with the help of on-screen prompts. You can interrupt the wizard at any time. The steps are not executed as a unit until all steps have been completely configured.

5.2.1 Licensing

All Emalytics hosts have a host ID on which licensing is based. Licenses are created, transported, and stored as license files. In addition to the host ID, each license file contains the Niagara version to which it applies, as well as a number of features that it enables. There are three possible license sources:

License files can be obtained via the sales channel and downloaded from the file system (offline).

All licenses are generated on a publicly accessible license server and can be accessed online by the Workbench at any time. Licenses can be updated on the license server by purchasing additional options (new features) or as part of software maintenance (higher version).

The Emalytics Automation Workbench simultaneously maintains its own local Workbench license database in which all licenses installed by the Workbench are automatically entered.
You can maintain this Workbench license database in the **“Workbench License Manager”** (**“Tools – Local License Database”** menu). Many licenses can be synchronized with the license server online so that they can later be installed offline in controllers via commissioning.

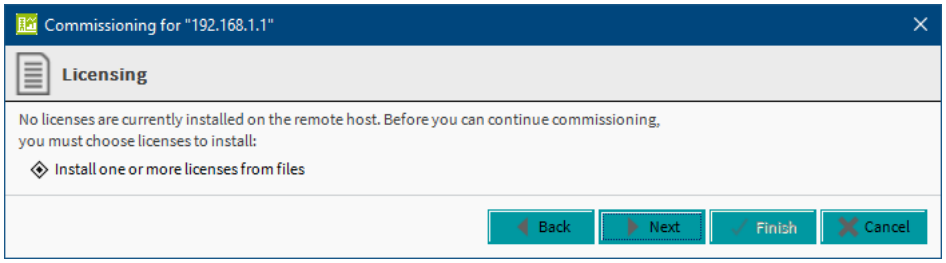


Figure 5-12 Commissioning Wizard: License installation

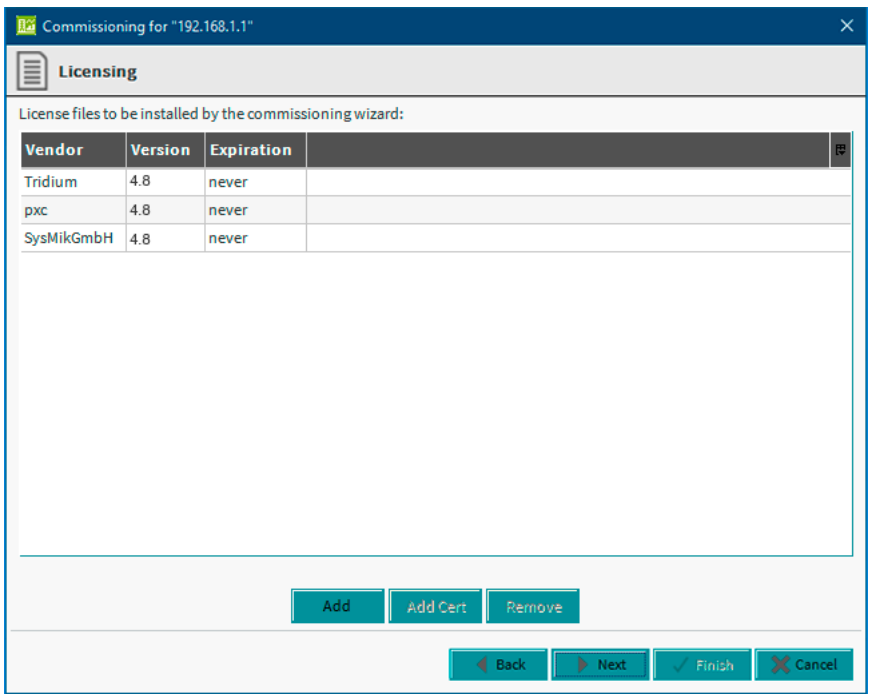


Figure 5-13 Commissioning Wizard: Available licenses are installed via the “Add” button.

5.2.2 Runtime profiles

The Emalytics Automation Framework has a modular structure. Functions are grouped into modules. Modularization is based on two criteria: the intended application and the runtime profile.

For example, for the intended application of trend data collection, there are three modules for different runtime profiles: *history-rt.jar*, *history-ux.jar*, and *history-wb.jar*.

The runtime profile selected during commissioning determines which of these modules should be installed:

The **“RUNTIME”** runtime profile contains the core functions without a user interface and is always enabled. The module for history collection is *history-rt.jar*.

UX and **WB** must also be enabled if you want the station to have a web visualization. These are user interfaces such as managers or plug-ins. For trend data collection, these are the modules *history-ux.jar* and *history-wb.jar*.

The **DOC** option, which contains the online documentation, is less common.
Trend data collection is documented in the *docUser-doc.jar* online user manual.

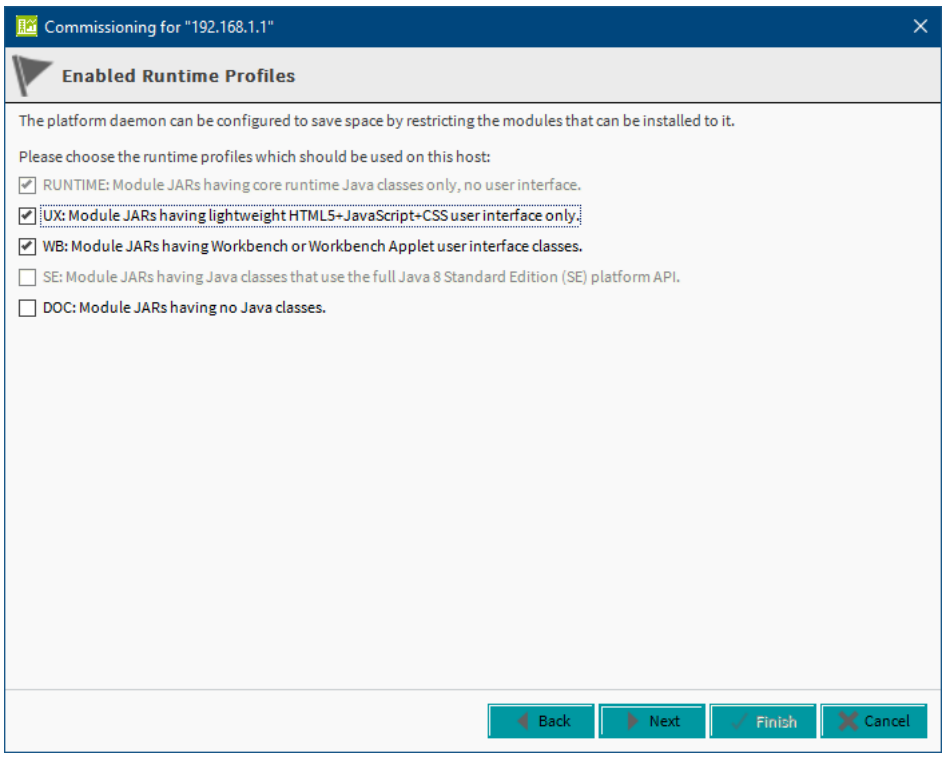


Figure 5-14 Runtime profile

5.2.3 Installing a station

If a station already exists in the file system of the Workbench, it can be installed in the host with this commissioning step.

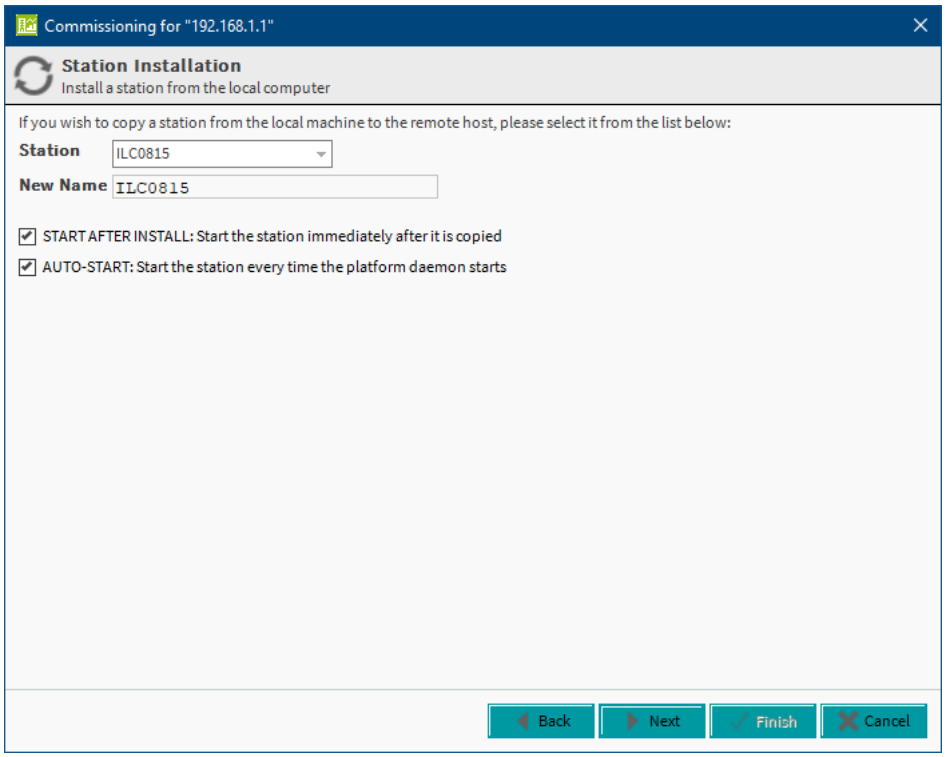


Figure 5-15 Station installation

5.2.4 Install the modules

In this commissioning step, all modules required to operate the host are automatically installed. These include a number of basic modules as well as modules that are required by the selected runtime profile (Section 5.2.2) or by the station (Section 5.2.3) to be installed. Module upgrades and other modules can also be installed manually for later use.

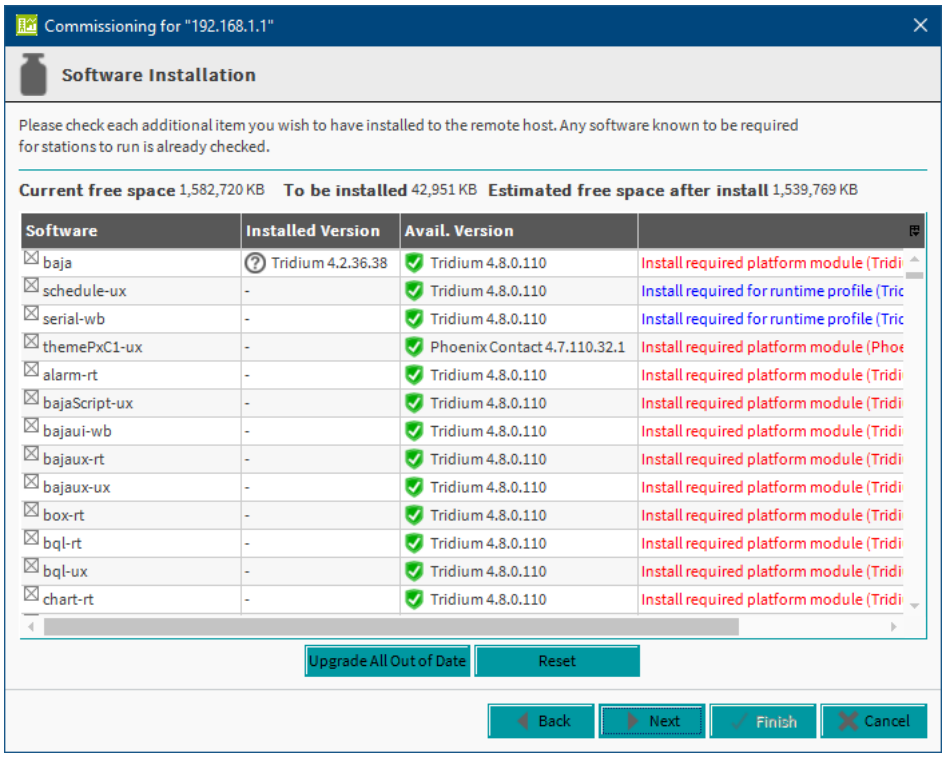


Figure 5-16 Installing software modules

5.2.5 Installing the system software

System software includes platform-specific software elements: the operating system, Java Runtime Environment, and Niagara Runtime Environment. These software packages are delivered and versioned in the form of distribution files. The distribution files for the ILC 2250 BI are part of the *Emalytics add-ons*. During commissioning, new versions of the system software are transferred to the host and installed there.

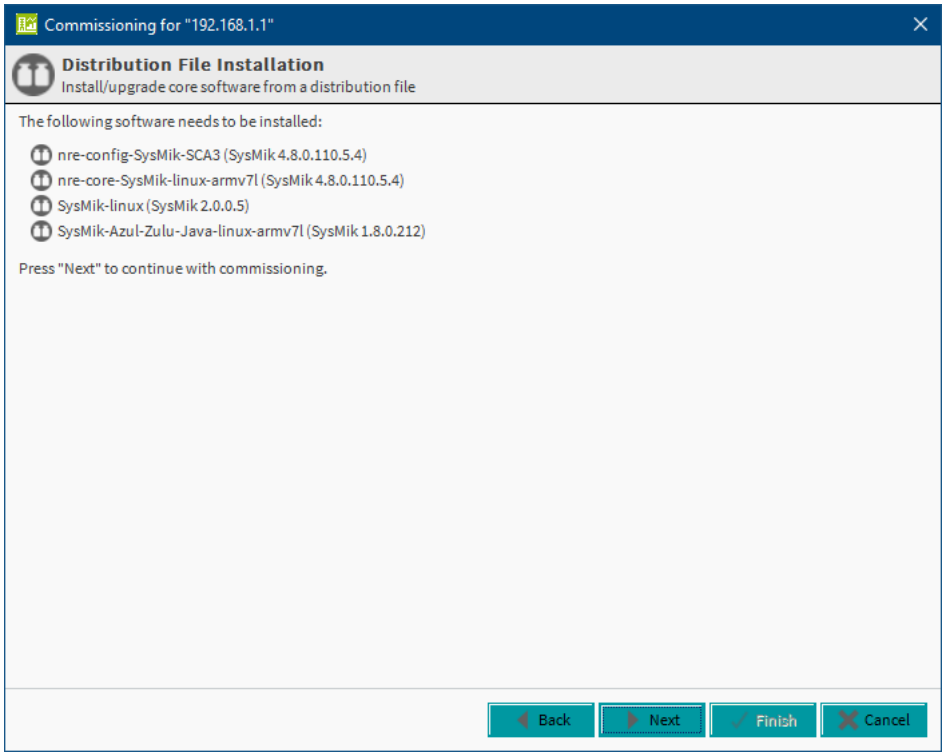


Figure 5-17 Installing the system software

5.2.6 Setting the time

During commissioning, the time of day and the time zone can also be transferred from the Workbench to the host. There is no input screen for this step. If you want to define a different time and time zone than the settings in the Workbench, you can do this outside the Commissioning Wizard using the **Platform Administration** platform tool.

5.2.7 IP settings

In this step, IP settings can be defined for all available network adapters of the host.

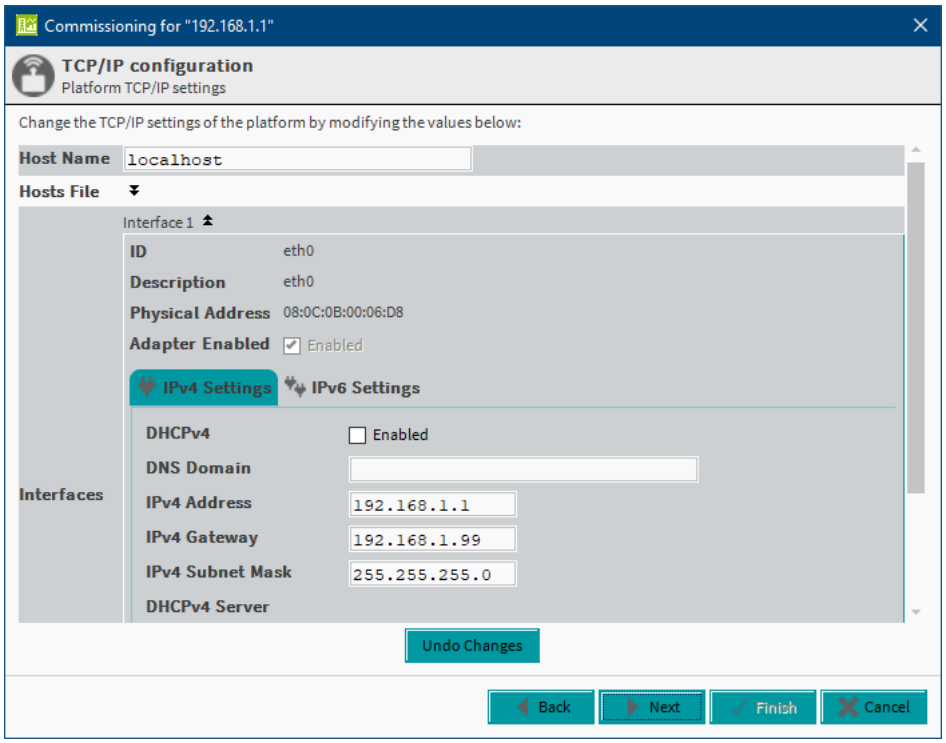


Figure 5-18 IP configuration

Note: If no gateway is used when configuring the network settings, the entry depends on the controller used:

- **With the ILC 2250 BI(-L), the gateway fields remain empty in this case.**
- For controllers of the predecessor series ILC 2050 BI(-L), the value "0.0.0.0" must be entered for the gateway.

5.2.8 Configuring platform users

In the delivery state, the platform user name and password for the ILC 2250 BI controller are “phoenix” and “inline” respectively. These are replaced by individual access data when the platform connection is established for the first time. You can create additional platform users during this step.

NOTE

The platform access data is important and must not be forgotten!

Store your access details securely. There is no way to reset the password without resetting the device to default settings.

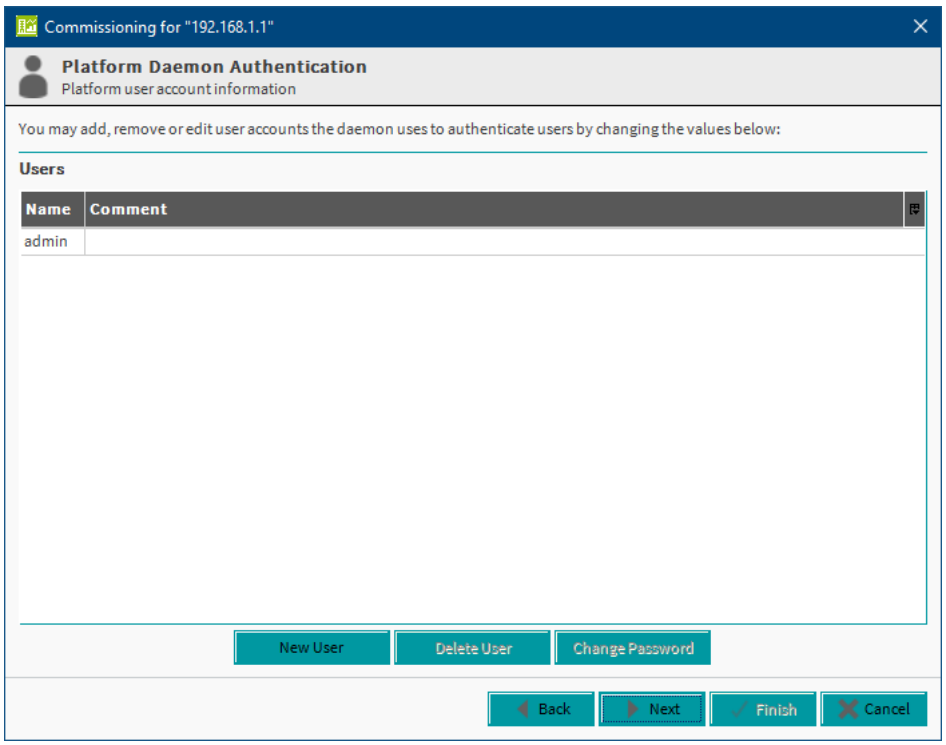


Figure 5-19 User configuration

5.3 Completing commissioning

All information recorded by the Commissioning Wizard is displayed again. You complete the commissioning process by pressing the Finish button. This can take a few minutes. The host is restarted after the files are loaded.

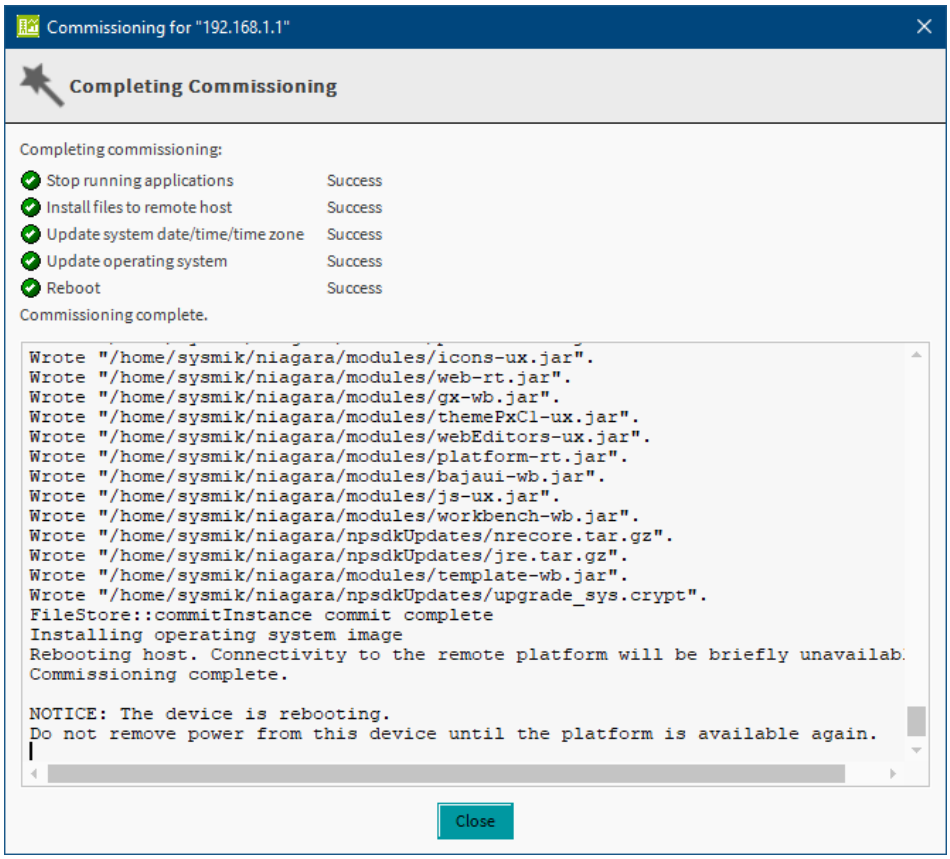


Figure 5-20 Completing commissioning

When the host restarts, any transferred system software is installed, and the Emalytics host will restart automatically several times.

The process status LEDs on the ILC 2250 BI indicate the current system status.

Table 5-1 Process status LEDs (marking)

LED	Meaning
PL	Status of the Emalytics Platform
ST	Status of the Emalytics Station
SE	Status of the I/O server

In normal operation, all three LEDs are activated according to the same scheme. It is very easy to see which software processes are currently starting (high processor load) or when a startup process has been completed (low processor load).

Table 5-2 Process status LEDs (function)

Process status LED	Meaning
Off	The process has not started.
Flashing yellow with 2 s cycle duration, 10% (on): 90% (off)	The process requires less than 10% of the processor time.
Flashing yellow with 2 s cycle duration, 50% (on): 50% (off)	The process requires between 10% and 50% of the processor time.
Flashing yellow with 2 s cycle duration, 90% (on): 10% (off)	The process requires more than 50% of the processor time.

In addition, during a software upgrade, the process status LEDs indicate the installation process with a running light effect.

Once all three LEDs flash in unison, the commissioning process is finished and the Emalytics Automation Station will run on the host.

6 Station engineering (tutorial)

Now that the Emalytics Automation host has been commissioned, online engineering of the station can begin. Engineering a station is essentially a type of graphical programming. Required modules are installed and components of different modules are created, configured, and connected to each other in the station.

For this activity, it is best to use the Workbench with the **“Nav”** side bar for navigating in the station and the **“Palette”** side bar for selecting components from modules. You can enable both side bars from the **“Window – Side Bars”** menu.



You can also engineer a station offline in the file system of the Emalytics Automation Workbench. However, no application functions are executed in this case.

By loading a station in the Supervisor, you can edit a station online without a controller. However, some functions such as I/Os and serial drivers are not available in the Supervisor. In addition, functions may be restricted by the Supervisor license.

6.1 Components

Components are the building blocks for stations. They are grouped into modules according to application. Components have slots. These can be properties, actions, or topics.

- An **action** is an input via which an action is initiated. Actions can be initiated via the local **“Run”** menu.
- A **topic** is an output via which an event is reported.
- A **property** can be many different things. Components can be properties of components. This is how the tree structure of the station is created. Properties with less complex data types are used as inputs and outputs, or parameters of the components.

6.2 Views

Components have views that show specific aspects of the component. There are various default views that, with a few exceptions, are available for all components.

- The **“AX Property Sheet”** view shows the visible properties of a component together with their value in a table. Writable properties can be modified here.
- The **“AX Wire Sheet”** view graphically displays all (lower-level) components of the component as a block diagram. Links between components can be created and deleted in this view.
- The **“AX Slot Sheet”** view shows all (even non-visible) slots of the component in a table. Via the local “Config Flags” menu, you can change flags that control the visibility of slots (*Hidden*), for example.
- The **“Property Sheet”** view is a combination of “AX Property Sheet” and “AX Slot Sheet” in HTML5.
- The **“AX Relation Sheet”** view shows relationships (including connections) of the component to other components.

Each component may also have additional specific views. For example, **“UserService”** (Station/Config/Services/UserService) has a specific **“User Manager”** view that lets you easily create and manage users.

All views of a component can be selected from the **“Views”** context menu.

6.3 Station connection

The Emalytics Automation Workbench is used for engineering a station. To do this, establish a new connection between Workbench and the host:

- **“File – Open – Open station”** menu.
- As session type, select **“Station SSL connection” or “Station connection”**.
- Enter the host IP address and confirm the input with **“OK”**.

All Workbench connections are stored in the **“Nav”** side bar in the nonvolatile memory. If the navigation tree already contains the IP address of the host, you can also create the station connection via the local menu of the **“Open Station”** IP address without having to enter the IP address again. If the navigation tree already contains a station connection, simply double-click to use the connection.

The public key of the host is displayed for confirmation upon initial contact with an Emalytics Automation host via a secure encrypted protocol (TLS). Afterwards, you log in with user name **admin** and password **Admin12345**, which was assigned when the station was created (Section 5.1.3). If you select the **“Retain login data”** checkbox, the login credentials are stored in the Workbench and no longer have to be entered in order to establish a connection.

Via the **“Tools – Manage Credentials”** menu, you can view, reset, and delete all login credentials (without passwords, of course) stored by the Workbench.

Once the connection is established, the **“Station Summary”** view, similar to the navigation tree, displays elements of the station as well as station properties.

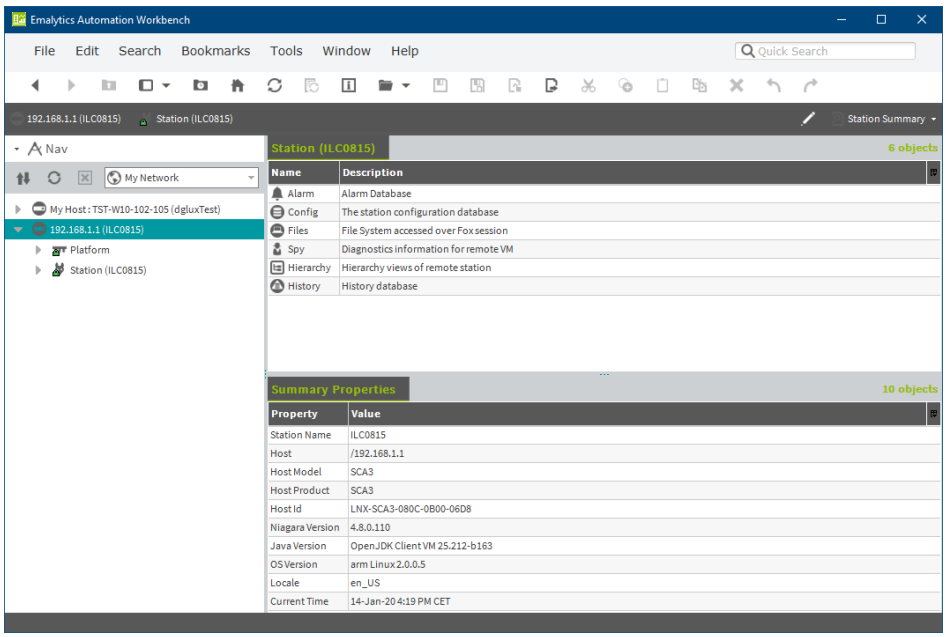


Figure 6-1 Station overview

At this point, we are only interested in the **“Config”** element. This is the engineering level of the station. All lower-level elements were saved as XML files when the new station was created (Section 5.1.3) and now “live” in the Java Virtual Machine of the host after the station has been installed. An example of this is the **“FoxService”** under **“Config / Services”**, which implements the Fox protocol for communication between the Workbench and station.

6.4 Hello world – first application

The first exercise is to add two numbers.

6.4.1 Installing the module

The **kitControl** module contains an add component. The module includes numerous primitive components that perform arithmetic or logical functions.

Before you can create components of a module in the station, you must install the corresponding module in the host. You do this using the **“Software Manager”** platform tool.

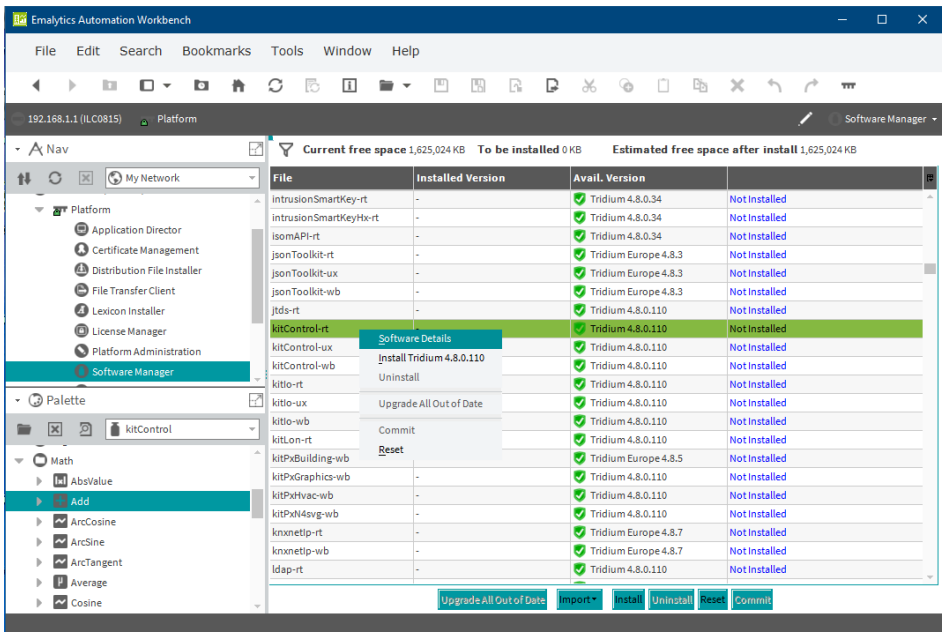


Figure 6-2 Software Manager workbench

The Software Manager displays all modules of the host (**Installed Version**) and the Workbench (**Avail. Version**). You can select **kitControl-rt** for installation via the local menu. Choose **“Commit”** to transfer the module to the host.

6.4.2 Creating the component

Use the local **“New – Folder”** menu in **“Config”** to create a new folder called **“MyAdd”**, in which the application is to be implemented. Double-click the newly created component to open the **“Wire Sheet”** default view.

The **“Palette”** side bar contains controls. When you move the cursor over these elements, an explanation is displayed at the bottom left of the Workbench. Using the **“Open Palette”** control, open a list of all modules of the Supervisor installation. Enter the first few letters of **kitControl** to quickly narrow the selection and then select **kitControl**. The components of the **kitControl** module are now displayed by category in the **“Palette”** side bar.

Create an additional component in the station by dragging the **“Add”** component from the **“Math”** folder in the palette to the **“Wire Sheet”** of the **“MyAdd”** folder in the workspace.

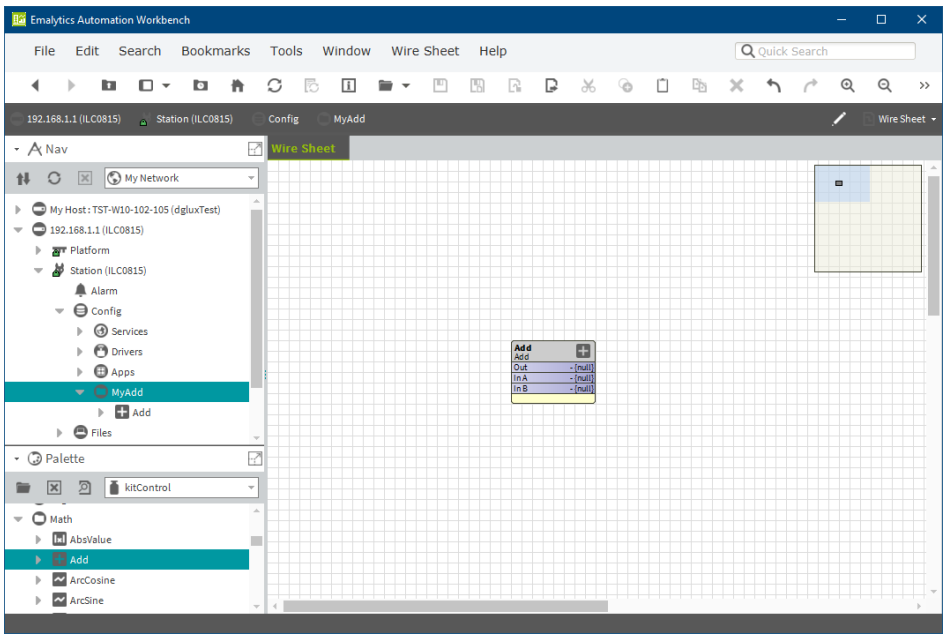


Figure 6-3 Workbench workspace

The content of the workspace can be described using an “Object Resolution Descriptor” (ORD). You can easily switch between the most recent views by choosing **“File – Recent Ords”** or **“File – Back”** and “**File – Forward”** from the menu.

6.4.3 Configuring the component

Double-click **“Add”** to open the default view for this component, **“AX Property Sheet”**. All properties of the component and their values are displayed in this view. Enter values for *InA* and *InB* and save your changes to the parameters by choosing **“Save”**. The *Out* property immediately displays the result of the add operation. The result is displayed in the Workbench. However, the station performs the calculation on the host.

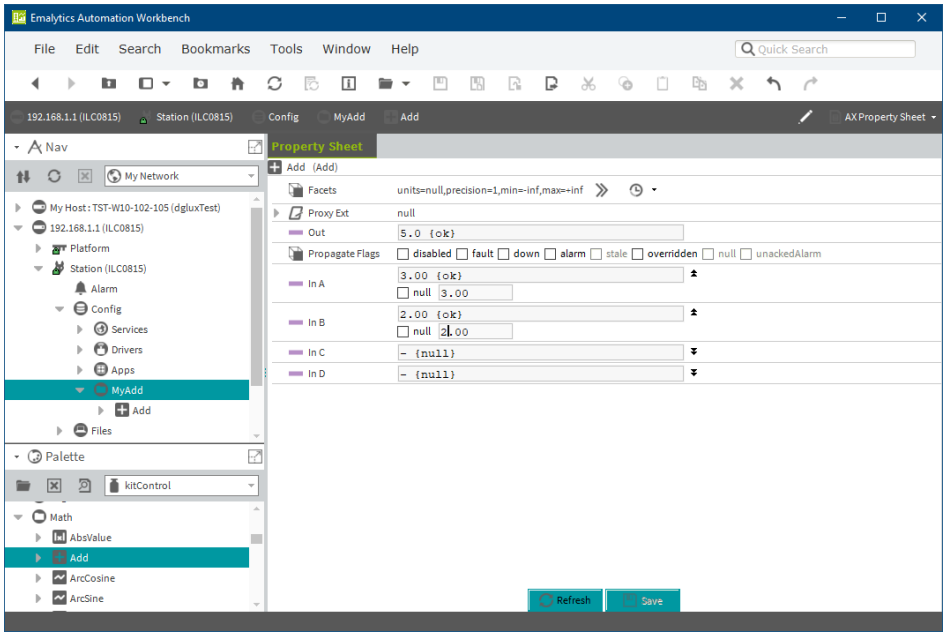


Figure 6-4 AX Property Sheet

6.4.4 Connecting components – links

You now want the add component to process a discrete signal (that is, a signal that changes its value at certain intervals) instead of a fixed parameter value.

- The **kitControl** module also contains signal generators. Insert a **“SineWave”** sine signal generator from the **“Util”** folder of the **kitControl** palette into **“MyAdd”**. With the default configuration, this component generates a sine signal with an amplitude of 50.0, an offset of 50.0, and a period of 30 seconds. The output value is calculated every second. This means that the output values fluctuate between 0.0 and 100.0.
- To connect the output of the signal generator to an input of the add component, move your cursor over the right third of the *Out* slot of the **SineWave** component in the Wire Sheet of **“MyAdd”**. The cursor will turn into an arrow (→).
- Now press the left mouse button and drag the cursor (now a closed fist) over the left third of the *InA* slot of the **“Add”** component (→) and let go of the mouse button.

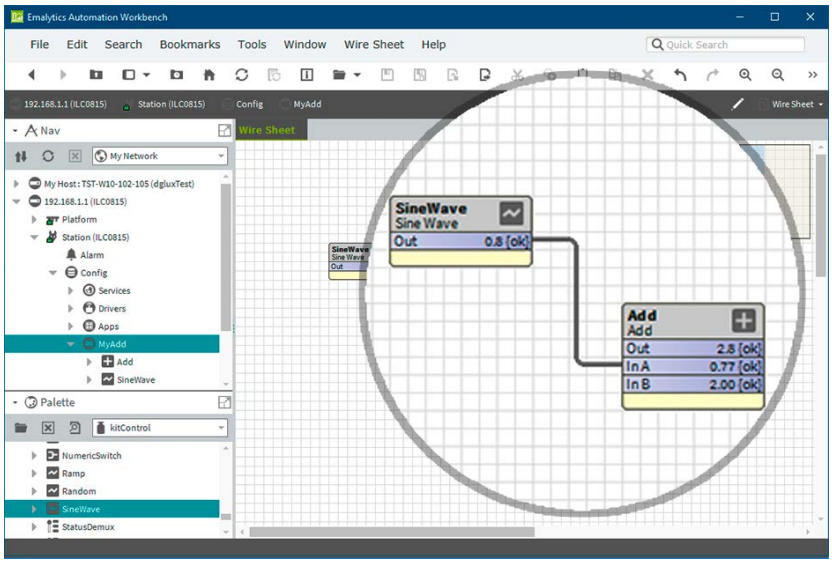


Figure 6-5 Connecting components

↪ A connection between **“SineWave.Out”** and **“Add.InA”** is established and displayed in **“Wire Sheet”**. You can now clearly see how the add component recalculates the output each time the input value is changed.

You can also interconnect components that are displayed in different structures and therefore cannot be displayed on a **“Wire Sheet”**.

To do this, first select a component from the local menu by choosing **“Link Mark”** for a connection and select **“Link From”** or **“Link To”** in the local menu of another component, depending on the desired direction of the connection. This opens a dialog window in which you can select the slots of both components to be connected.

The **“Relation Sheet”** view of a component displays a table of all connections of the component. These connections can also be deleted here via the local menu.

6.5 Points

Niagara uses points for the purpose of normalizing information. Points are the basis for all communication drivers as well as for many other functions such as trend data collection, alarm management, and scheduling.

There are 4 types of points: *Numeric* (floating-point number), *Enum* (lists), *Boolean* (true/false), and *String* (text). A distinction is also made between writable points and read-only points. All point types are contained in the **“kitcontrol”** palette.

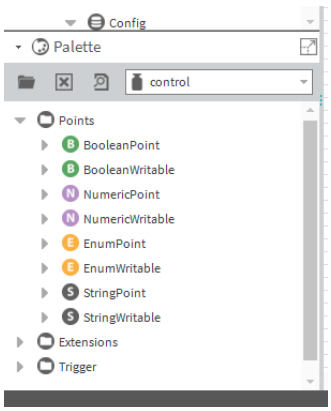


Figure 6-6 Point palette

6.5.1 Point value and status

Each point has an *Out* output, which also contains a *Status* in addition to the point value of the respective data type. Important general statuses are:

– **{ok}** – everything OK
– **{null}** – the point does not contain a value; important for priority control
– **{overridden}** – the point is manually overridden
– **{alarm}** – the point meets an alarm condition
– **{unacked alarm}** – the alarm is still unacknowledged.

6.5.2 Writable points

Writable points contain a prioritization mechanism: There are 16 inputs corresponding to 16 priorities. The *Out* point value is defined by the highest-priority input with a signal that is not equal to {null}. Without an input signal, the output goes to *Fallback*. This allows a writable point to be accessed from multiple sources, for example heater control via timer, presence signal, manual override, frost protection, and fire indicator panel.

Writable points have actions that allow the points to be manually overridden. These actions are also available later in the web visualization. The **fallback** can be set by choosing **"Actions – Set"** in the local menu. The **"Manual mode"** and **"Automatic mode"** actions can be used to control or release the *In8* input, whereas **"Emergency manual mode"** and **"Emergency automatic mode"** are used to set *In1* to the highest priority.

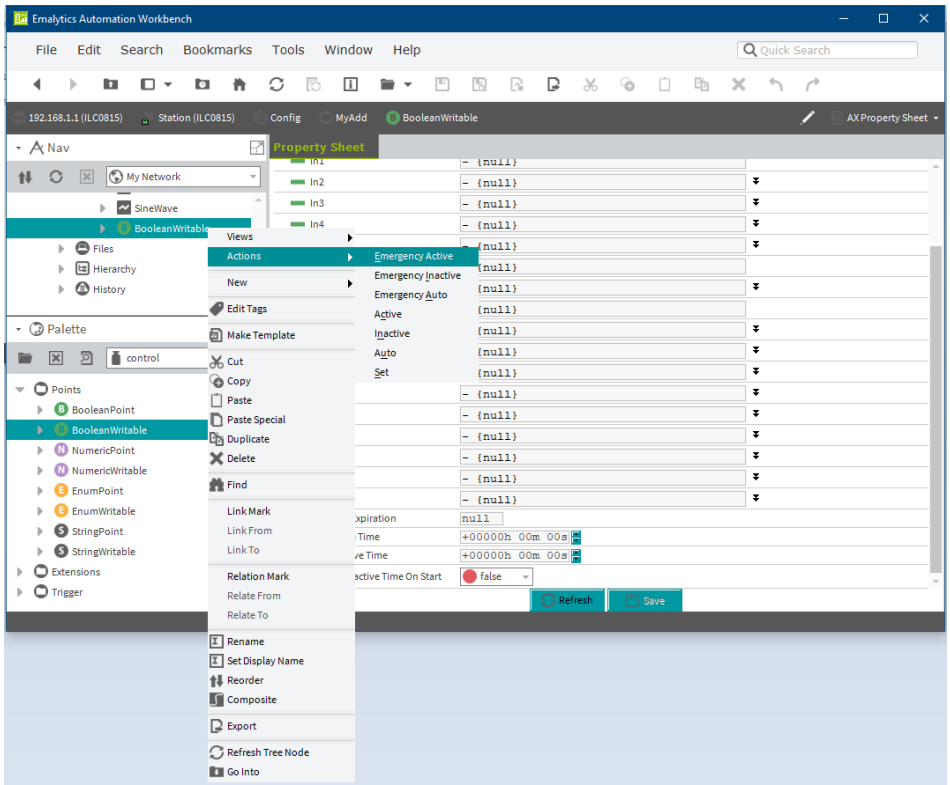


Figure 6-7 Manual/Emergency/Automatic mode menu structure

- Create a writable point in the **"MyAdd"** folder by selecting **"New – NumericWritable"** from the local menu.
- Double-click to open the **"AX Property Sheet"**.
- Change the input values and check how priority control works.
- Connect the *Out* output of the point to *InB* of the add component.

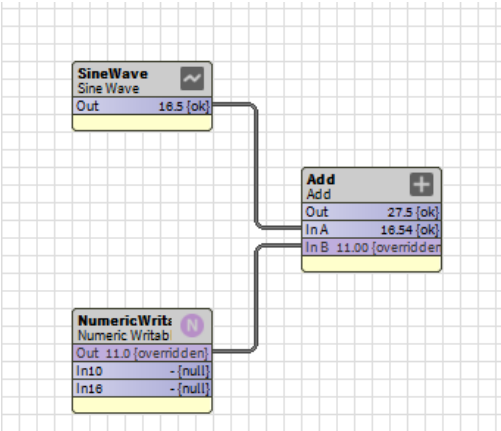


Figure 6-8 Writable points

- Overwrite the point value by choosing **“Emergency Overwrite”** from the local menu.
- You can deactivate the override again with **“Emergency Auto”**.

Slots of components have a **“Summary”** flag that controls their visibility in the Wire Sheet. Slots without this flag are only displayed if they are connected to another slot. These flags can be changed via the **“Pin Slots”** local menu.

6.5.3 Read-only points

Read-only points have an *Out* output whose value is determined by an internal function and typically cannot be overridden directly by inputs or actions. Many components with processing functions are special derivatives of read-only points. For example, **“Sine-Wave”** and **“Add”** from the **kitControl** module are read-only **“NumericPoints”**.

Read-only points from the control module have no function and are therefore of no significance for station engineering.

6.5.4 Facets

Facets are used to control how points are represented on the user interface.

- To expand the **“AX Property Sheet”** of **“Add”**, click the plus icon in the line for the *Facets* property.
- Set the *precision* facet to 2 to display the *Out* output value with two decimal places.
- In the *units* facet, set the unit to “power / kilowatt” to display the **Out** output value with the physical unit kW.

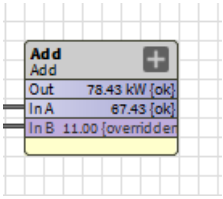


Figure 6-9 Property facet

In the **“AX Property Sheet”** of a point, you can simply use the icon of the *Facets* property to assign one of your recently used facet definitions instead of always having to assign the same facets one at a time for multiple points.

6.6 Trend data collection – history

Trend data collection is a standard function of the Emalytics Framework. This data is always collected with relation to points. Collection can take place at regular intervals or it can be event-driven when changes are made (COV = change of value). The depth of trend data is configurable. Trend data can be displayed as part of the web visualization. Typically, trend data may also be cyclically transmitted to a higher-level Emalytics Supervisor for more extensive analyses/visualizations or archived in databases via database drivers of the Supervisor.

Trend data is collected by components of the **“history”** module. These components are not connected to the points via links. Instead, they are placed below the point as **“Point Extensions”**. In other words, the history extension becomes a slot of the point.

- Open the **“history”** palette.
- Double-click the **“NumericInterval”** component in the **“Extensions”** folder of the **“history”** palette to display the **“AX Property Sheet”** view of the palette component.

↪ You can now configure component parameters before they are applied to the station.

NOTE: Parameter changes in the palette are not saved.

After the Workbench is restarted, all parameters in the palette return to their default values.

Trend data is saved under a name that is defined by the *History Name* property (expand “History Config”) of the history extension. The default value *%parent.name%* is a reference to the **“name”** slot of the higher-level component.

The capacity of a record can be set via *Capacity*. The default value of 500 in combination with the default interval of 15 minutes ensures that history is recorded over 5 days. Emalytics Controllers have limited resources and are not suitable for long-term archiving.

Set an *Interval* of one second for real-time trend data collection.

- Extend the station tree in the **“Nav”** side bar so that you can see the components in **“MyAdd”**.
- Using drag and drop, insert one **“NumericInterval”** component configured in the palette into each of the **“Add”** and “**SineWave”** components in the **“Nav”** side bar.

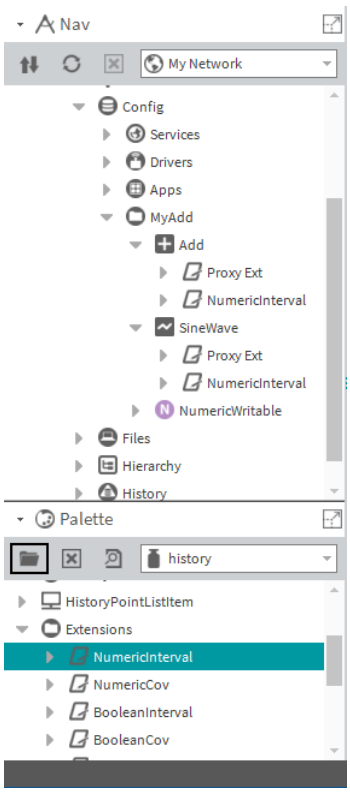


Figure 6-10 Navigation bar

The concept of point extension makes it very easy to copy a configured extension of a point (local **“Copy”** or **“Paste”** menu) to other points from the clipboard.

You must now start trend data collection. For individual points, you can do this via the *Enabled* property of the “History Extension”. The default view of the **“HistoryService”** under **“Config / Services”** is better suited for synchronized control of numerous trend data records.

- In the table of all history extensions, select both lines and start trend data collection by choosing **“Activate Histories”** from the local menu.

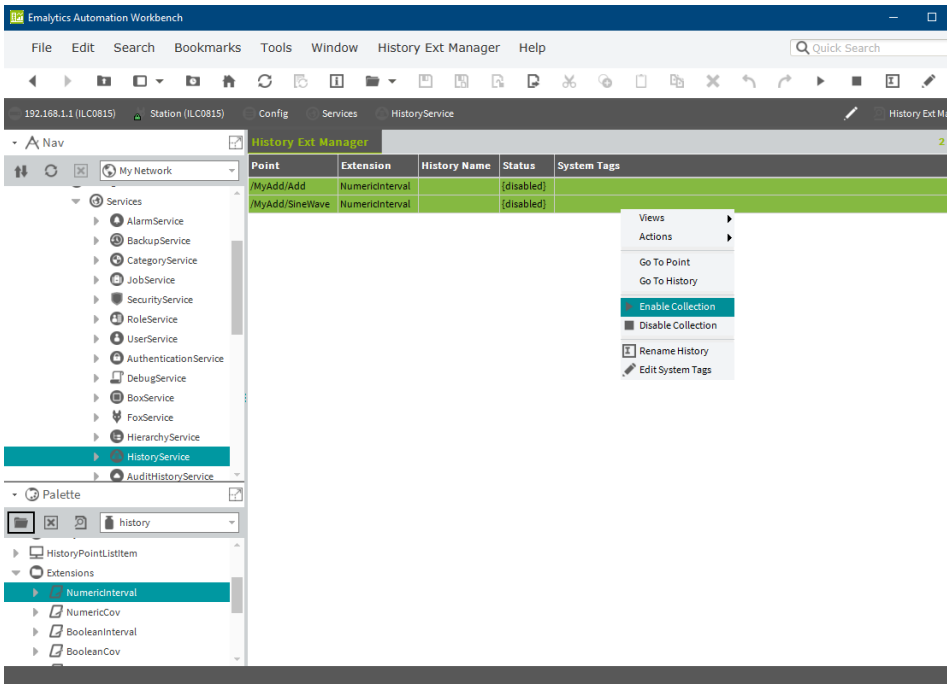


Figure 6-11 History extension manager

You can view the trend data via **“Station / History”**.

- Double-click **“Station / History / ILC0815 / Add”** in the **“Nav”** side bar to display the **“Add”** trend data of the **“Add”** point in the **“Chart”** default view.
- Drag **“Station / History / ILC0815 / SineWave”** from the **“Nav”** side bar to the workspace to display both trend data in one diagram.

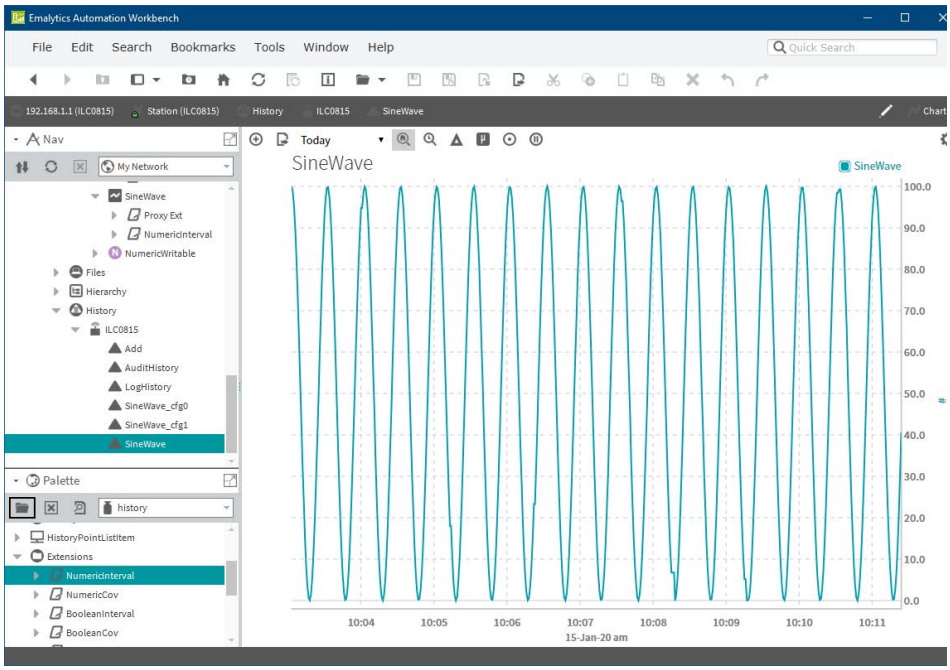


Figure 6-12 Trend data collection

NOTE

Trend data is saved in the nonvolatile memory. Emalytics Controllers use flash memory for storage, which can only handle a limited number of write cycles. Continuous collection at frequent intervals can damage the flash memory.

- Stop trend data collection via the **“HistoryService”**.

6.7 Alarm management

The Emalytics Automation Framework features comprehensive alarm management.

Alarm sources generate alarms and route them to alarm classes. Alarms can be routed from alarm classes to different alarm recipients. Typical alarm sources are points that contain an alarm extension as well as network and device components of drivers.

Typical alarm recipients are alarm consoles as part of a web visualization, emails, text messages, or printers. Alarms are routed to higher-level Supervisor stations so that they can be centrally processed. Alarm classes have built-in escalation management. This means that pending alarms can be routed to different recipients with specific delays.

Alarms are routed via the **“AlarmService”** under **“Config / Services”**. This service already contains the **“DefaultAlarmClass”** to which all alarm sources send their alarms.

- Create a new **“MyAlarmClass”** by dragging the **“AlarmClass”** from the **“alarm”** palette to the “**AlarmService”**.
- For the alarm recipient, insert **“ConsoleRecipient”** from the **“Recipients”** folder of the **“alarm”** palette into the **“AlarmService”**.
- In the “**Wire Sheet”** of the **“AlarmService”**, connect the **“Alarm”** topic from **“MyAlarmClass”** to the **“Route Alarm”** action of the **“ConsoleRecipient”**.

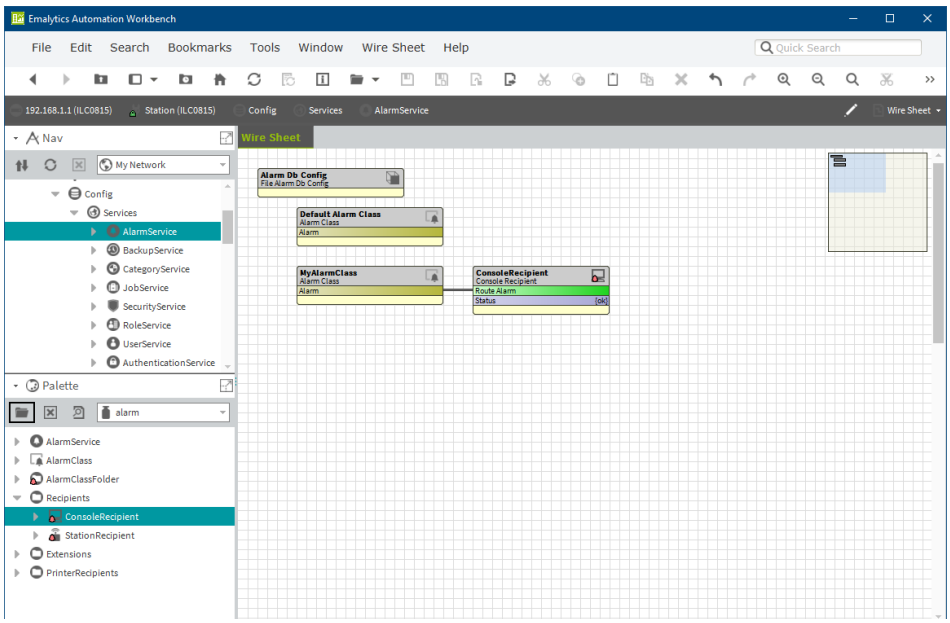


Figure 6-13 Wire Sheet – alarm console

- Extend the station tree in the **“Nav”** side bar so that you can see the components in **“MyAdd”**.
- Using drag and drop, insert an **“OutOfRangeAlarmExt”** component from the **“Extensions”** folder of the **“alarm”** palette in **“Add”** into the **“Nav”** side bar.

↪ This alarm extension can monitor the value of a numeric point to ensure that it does not exceed limits. An alarm is generated if limits are exceeded.

Double-click the inserted alarm extension to configure it:

- Select *MyAlarmClass* for the *Alarm Class* property.
- Set *Offnormal Algorithm / Alarm limit high* to “80” as the upper alarm limit.

- Set *Offnormal Algorithm / Alarm limit low* to “20” as the lower alarm limit.
- Enter the following alarm text: *Offnormal Algorithm / High Limit Text* = “too high” and *Offnormal Algorithm / Low Limit Text* = “too low”
- Enable both alarm limits in *Offnormal Algorithm / Limit Enable*.

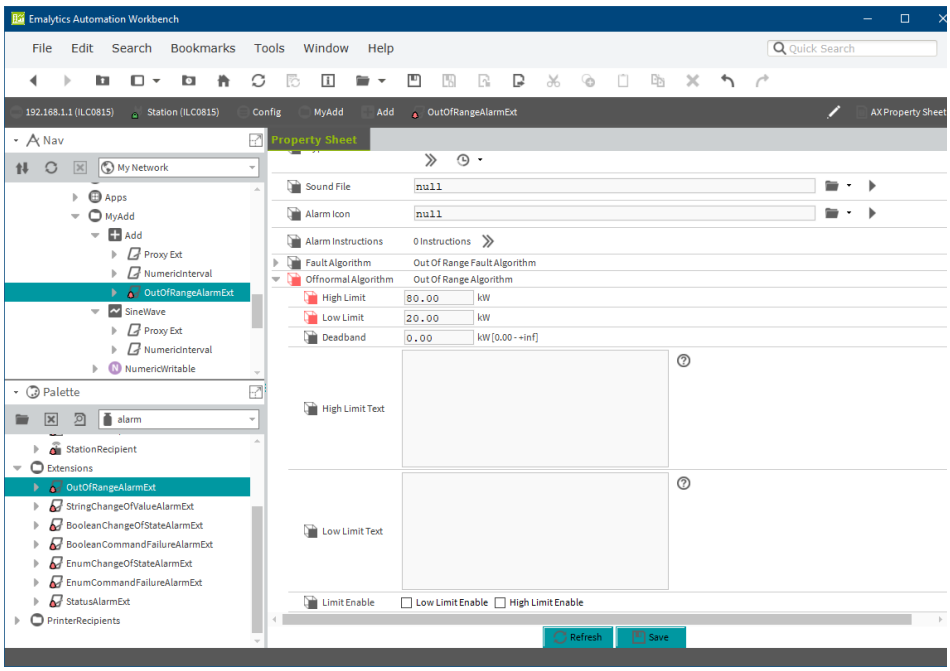


Figure 6-14 Property sheet for Out Of Range alarm

- Double-click **“ConsoleRecipient”** in **“Config / Services / Alarm Service”** to enable the alarm console, which now displays cyclically recurring alarms.
- Double-click an alarm to display a table of all alarm events for this alarm source.
- Double-click an alarm event for a detailed view with a precise record of when and why the alarm was triggered and when the alarm state was exited.

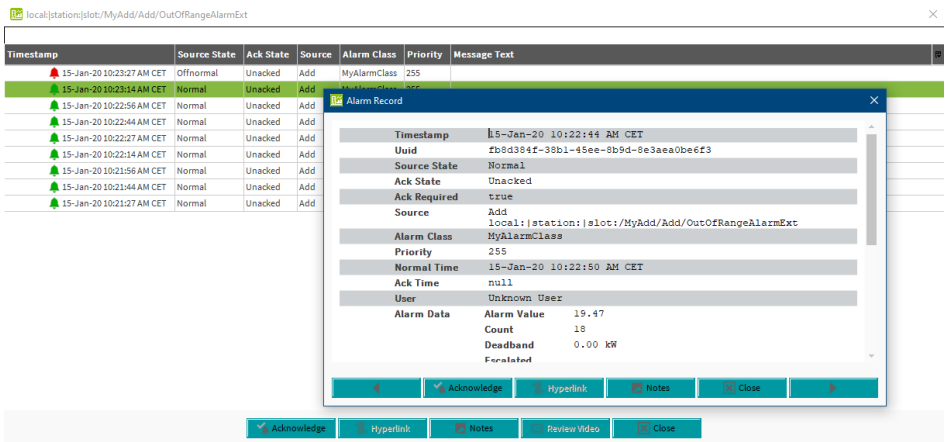


Figure 6-15 Acknowledge alarms

Alarms appear as active alarms (red bell) in the alarm console as soon as the alarm condition monitored by the alarm source is met and the generated alarm has been routed to the alarm console via the Alarm Service. Alarms become inactive (green bell) if the alarm condition is no longer met but the alarm has not yet been acknowledged. Events that require acknowledgement can be specified in the alarm class. Individual or entire groups of alarms can be acknowledged on all levels of the alarm console. A yellow bell indicates acknowledged active alarms. Acknowledged inactive alarms are not displayed in the alarm console.

The *capacity* of the alarm database can be specified via **“Alarm Db Config”** in the **“AX Property Sheet”** of **“Station / Config / Services / Alarm Service”**. The alarm database itself can be reached via **“Station / Alarms”**. The most recent alarms of the station are stored here depending on the configured capacity. The entries can be displayed and deleted here.

6.8 Scheduling

Scheduling in the Emalytics Framework is handled by components of the **“schedule”** module. The **“CalendarSchedule”** component has a Boolean output that is activated on configurable days. **“NumericSchedule”** has a numeric output that is controlled via configurable daily schedules for each day of the week as well as exceptions.

- Insert a **“NumericSchedule”** component from the **“schedule”** palette into **“MyAdd”** and connect the *Out* slot to the *In16* slot of *NumericWritable*.
- Double-click *NumericSchedule* to open the **“Schedule”** manager.
- In a weekly schedule, select a time range by moving the cursor to the start time, pressing the left mouse button, moving the cursor to the end time, and releasing the mouse button.
- In the *Event Output* field, enter the value to be displayed in the schedule for this time range.
- Make other entries in the schedule and copy them to other days using the local menu.

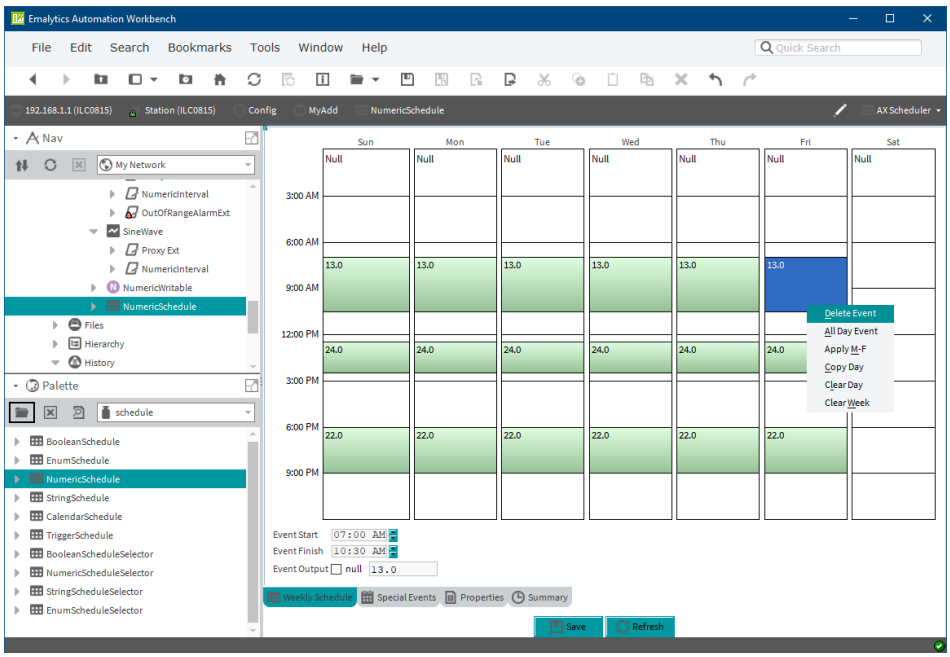


Figure 6-16 Calendar / Schedule workbench

↪ In the **“Special Events”** tab, you can configure exceptions (such as public holidays) and special daily schedules for these holidays that deviate from the weekly schedule.

↪ Click the “Add” button to open the **“Configuration”** window. Here, you can limit the validity period of the schedule and you can define the default output value of the **“Default Output”** schedule.

6.9 Web visualization

The Emalytics Station has a web server that can be configured via **“Station / Config / Services / WebService”**. You can connect to this web server via a web browser. To do this, you must sign in to the station with valid credentials. Depending on your user settings, you can access views and data of the station. This can range from basic visualization pages for monitoring a process to a comprehensive web workbench.

NOTE

The *WB* and *UX* runtime profiles must be enabled as a prerequisite for web visualization. Only then will the required modules and visual elements be installed in the Niagara platform.

The runtime profiles can be activated during the commissioning process (Section 5.2.2) or via the **“Configure Runtime Profiles”** button in the **“Platform Administration”** platform tool.

6.9.1 User-defined views

Each component has views. These can be default views like the **Wire Sheet** or component-specific views like driver managers. In addition to these programmed views, which are part of Emalytics modules, user-defined views can be configured during station engineering. These user-defined views are comprised of widgets. **Widgets** are graphical elements of the user interface that can interact with the station.

- In the context menu of MyAdd, select **“Views – New View”** to create a new view for this folder.
- Assign “**MyAddView”** as the **“View Name”**. This automatically derives the name of the PX file in which the design of the view is stored.

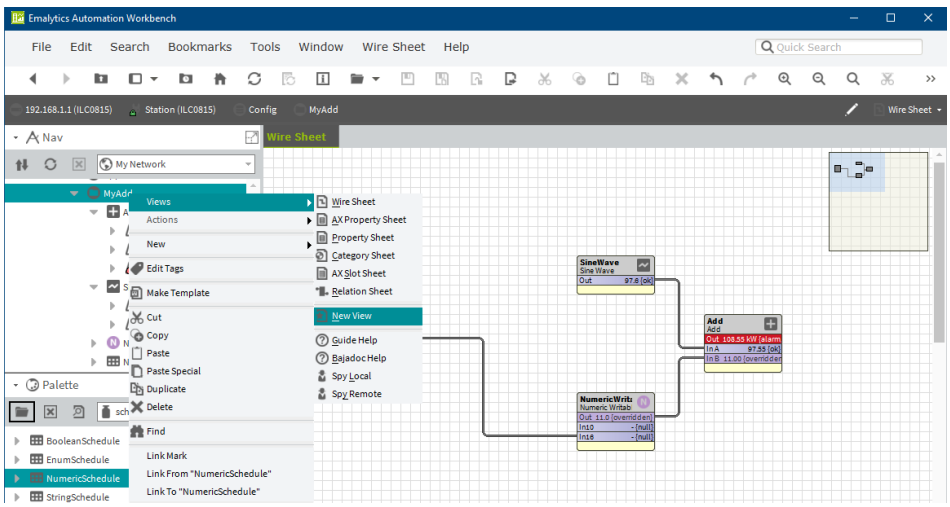


Figure 6-17 Creating new views

By storing the design in a PX file, the view can be used for multiple components once it has been configured.

6.9.1.1 PxEditor

The **“PxEditor”** of the new **“MyAddView”** view is displayed in the Emalytics Automation Workbench. Here, new widgets can be added to the view or existing widgets can be configured. The view is displayed in the workspace of the PxEditor. The different widgets can be selected, moved, or deleted.

The side bars on the right side of the PxEditor help you create the view. These side bars can be enabled or disabled using the button.

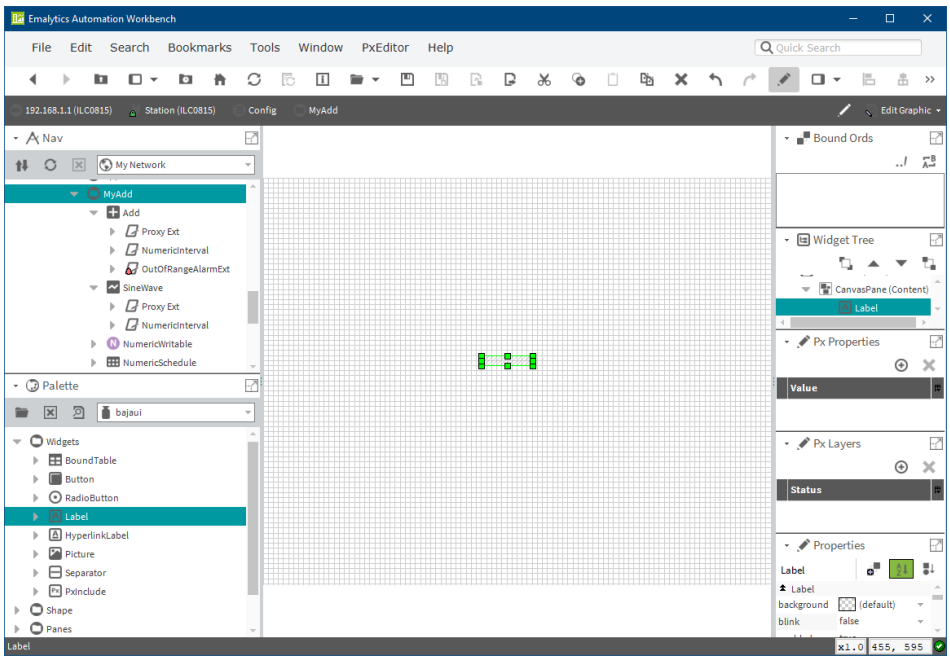


Figure 6-18 Px Editor with side bar

– The **“Bound Ords”** side bar contains a list of all connections of the view to the station via **“Ords”** (object resource descriptors).
– The **“Widget Tree”** side bar displays all widgets of the view in a tree structure. This side bar provides easy access to widgets, especially for more complex views with multiple panes and superimposed widgets.
– The **“Px Properties”** side bar displays the properties of the selected widget and allows you to configure them. Double-click a widget to open the **“Properties”** window, which has the same function. This window is more effective for engineering if you have a smaller screen.

6.9.1.2 PxViewer

Double-click **“Config / MyAdd”** to display the new, empty **“MyAddView”** view in **“PxViewer”** display mode. You can switch between the PxViewer and the PxEditor at any time via the **“PxEditor / Switch between display/edit”, “Hyperlink to pxEditor”** or using the corresponding button .

6.9.1.3 Widgets

Widgets are graphical elements of the user interface. Process data of the station can dynamically control (animate) properties of the widgets via configurable bindings. Widgets can also adopt user entries, thereby influencing the data of the station.

6.9.1.4 Manually configuring a basic widget

The **“Widgets”** folder in the **“bajaui”** module contains universal basic widgets of the Em-alytics Framework. The module also includes geometric primitives in the **“Shapes”** folder, as well as **“Panes”**, which control the layout of the view.

Adding a widget

- Enable the **“PxEditor”**.
- Drag and drop the **“Label”** widget from the “**Widgets”** folder of the **“bajaui”** palette to the workspace in the view.

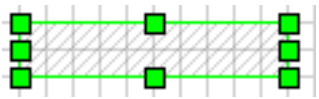


Figure 6-19 Adding a widget

Static configuration of a widget

- Double-click the widget in PxEditor to display the properties of the **“widget”**.
- Enter the character string “MyAdd” for **“text”**.



Figure 6-20 Configuring a widget

Widget animation – creating a binding

The next step is to animate some properties of the widget according to the value of a point of the station. To do this, you must first create a binding to the point.

- Click the **“Add Binding”** icon in the **“Properties”** window and select **“ba-jaui:ValueBinding”**.
- Click the browser button [...] of the **“ord”** field under the **“ValueBinding”** label to enter an object resource locator for a point.
- Click the expand arrow on the icon (right) to use the **“Component Chooser”** tool to select the **“Config / MyAdd / Add”** point in the station tree.

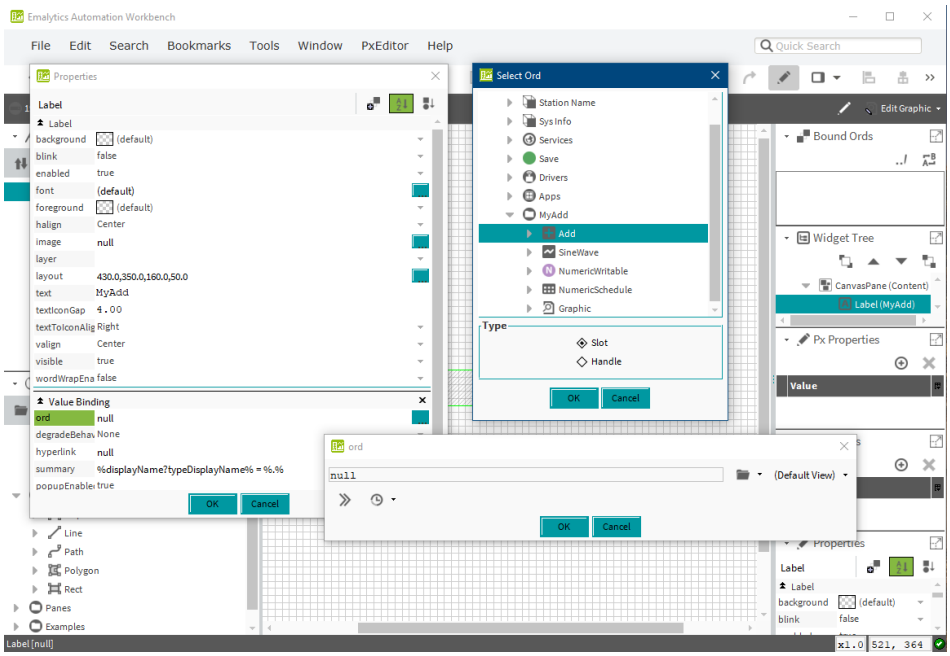


Figure 6-21 Widget binding

Animating a widget property

To change the background color depending on the point value, select the **“Animate”** menu item in the context menu under the **“Label”** label. Insert several value ranges (Numeric) and define different background colors (Simple).

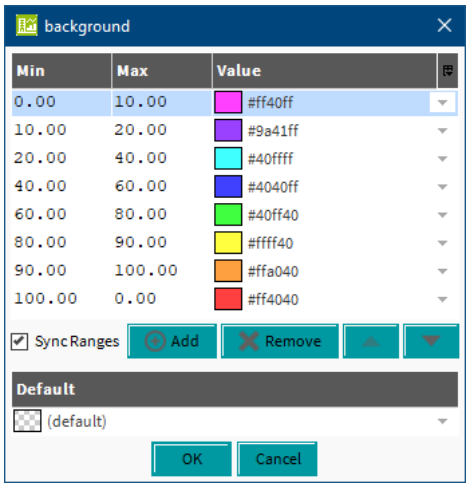


Figure 6-22 Defining widget properties

As a result, the background color changes depending on the value of the add component.



Figure 6-23 Modified widget

All properties of a widget can be animated in this way.

Animating character strings

The **text** property is a static “MyAdd” text with a unique reference to the **“Station / MyAdd”** folder. If you want to use a widget multiple times, it is best to animate and relativize all references to the station.

- Animate **“text”** via the **“Animate”** context menu by specifying the formatting rule for the text to be displayed. To do this, enter **“BFormat** string **%parent.name% = %out.value%”** in the **“Object-To-String”** converter type.

↪ When the widget is displayed, the BFormat string is converted: All elements enclosed by % signs are replaced by the values of the enclosed properties of the point specified by the binding.

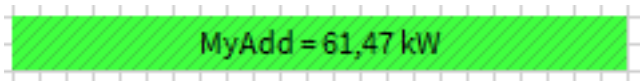


Figure 6-24 Animated BF string

Relativizing bindings

↪ The **“Bound Ords”** side bar of the **“PxEditor”** displays all references to the station that are used in widget bindings. It currently displays absolute Ords with the **station** as the reference point.

- Click the **“Relativize Ords”** icon ../ to convert these absolute Ords into relative Ords (the component of the station to which the view is assigned becomes the reference point).
- To see the resulting advantage, copy the **“Config / MyAdd”** folder in **“Nav”** using the local **“Copy”** menu and paste a copy in **“Config”** using the local **“Paste”** menu.
- Since the name **“Config / MyAdd”** already exists, a new name, **“MyAdd1”**, is proposed. Use this name.

↪ You now have two copies of your addition folder. Both have a user-defined view based on the same px file. Because of the relative Ords, the binding points to the respective lower-level **“Add”** point and the corresponding folder name is displayed correctly.

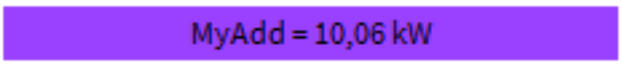


Figure 6-25 Absolute Ords

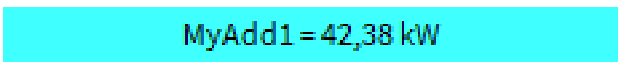


Figure 6-26 Relative Ords

6.9.1.5 Partially configured widgets

The user interface can be fully configured manually, although this is quite complex. Preconfigured widgets are already available for common visualization elements.

The “**kitPx”** module contains universal widgets and already includes default bindings and animations.

- From the **“kitPx”** palette, add a **“BoundLabel”** widget to the workspace of the PxEditor. This widget already contains a binding and an animated *text* property.
- You only need to specify the Ord of a point here in order to implement a simple value display.
- Click the browser button ... of the **“ord”** field under the **“Bound Label Binding”** label to enter an object resource locator for a point.

↪ Click the expand arrow on the icon (right) to use the **“Component Chooser”** tool to select the **“Config / MyAdd / SineWave”** point in the station tree.

6.9.1.6 Configuring widgets with wizard assistance

There is an even easier approach:

- Drag and drop the writable point **“Config / MyAdd / NumericWritable”** from **“Nav”** to the workspace of the PxEditor.

↪ The Ord is now already defined. The **“Make Widget”** wizard opens and displays all available visualization options for selection on the left side. On the right side, the widget resulting from the selected options is displayed with all of its properties.

- Select **“Bound Label”** on the left side and activate the **“Format Text”** and **“statusEffect (Color And Blink)”** options.
- Choose **“OK”** to include the value display in the user-defined view.

↪ The status of the point is animated by the background color. Since this is a writable point, the widget automatically provides the actions for manual override via a local menu.

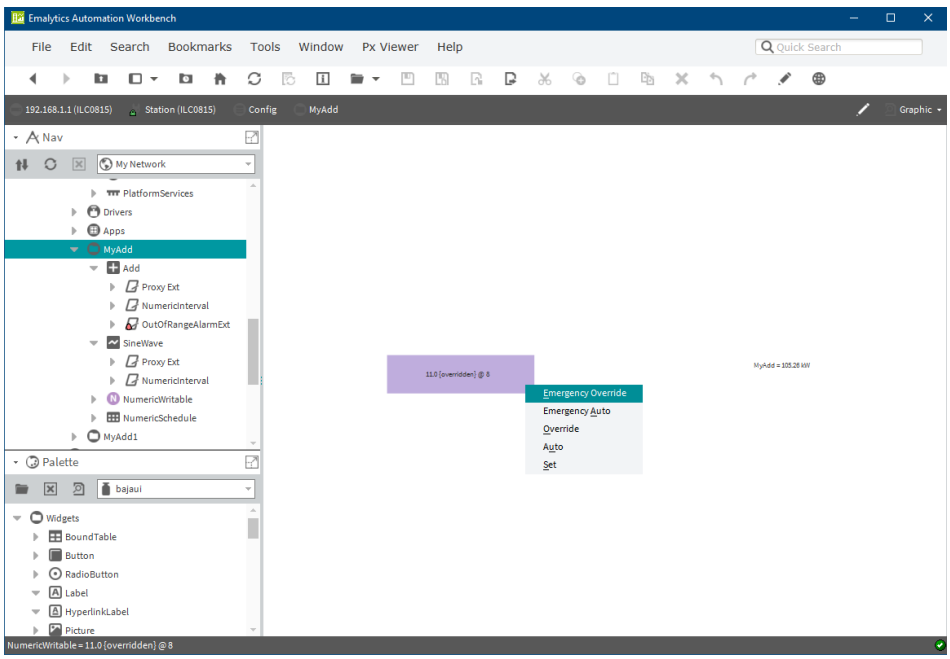


Figure 6-27 Widget with wizard

- Drag the writable **“Config / MyAdd / NumericWritable”** point into the PxEditor workspace. Select **“From Palette”** in the **“Make Widget”** wizard and the **“SetPoint-Slider”** widget from the **“kitPx”** palette to insert a value entry for the **“Fallback”** of the point via a slider into the user-defined view.

↪ However, for inherent reasons, this entry does not have an effect on the *out* point value if the point is overridden by the schedule with *priority 16,* or manually with *priority 8*.

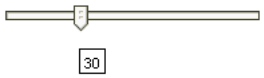


Figure 6-28 Widget wizard – slider

6.9.1.7 Visualizing system functions

- Drag the **“Config / MyAdd / NumericSchedule”** schedule into the workspace of the PxEditor and select **“Workbench View”** and **“Current Day Summary”** in the **“Make Widget”** wizard in order to insert the schedule of the current day into the user-defined view.

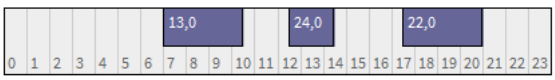


Figure 6-29 Visualized calendar function

↪ The widget has a hyperlink to the default view of the schedule. Of course you can always make changes to the widget configuration.

- Set the *hyperlinkActive* property to *false* to disable the hyperlink.
- Drag the **“Config / MyAdd / Add / Numeric Interval”** history extension into the workspace of the PxEditor and select **“Workbench View”** and **“History Chart”** in the **“Make Widget”** wizard in order to display the collected trend data in the user-defined view.

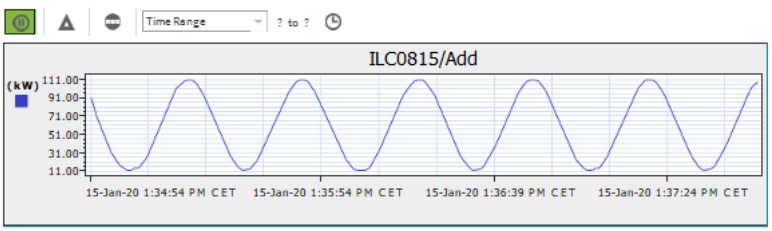


Figure 6-30 Widget with history chart visualization (time range)

- Drag the **“Config / Services / AlarmService / ConsoleRecipient”** alarm console into the workspace of the PxEditor and select **“Workbench View”** and **“Alarm Console”** in the **“Make Widget”** wizard in order to create the alarm console in the user-defined view.

6.9.1.8 Widgets for building automation

The **“kitPxHvac”**, **“kitPxGraphics”**, and **“kitPxN4svg”** modules contain specific building automation widgets that allow you to quickly create attractive system visualizations. Important system components such as boilers, fans, pumps, valves, registers, sensors, air ducts, and pipelines are assembled from the modules and animated by points of the station.

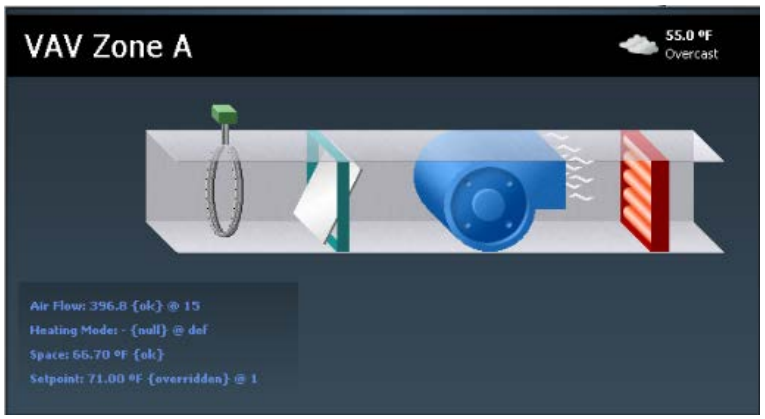


Figure 6-31 HVAC kit graphic (example)

6.9.2 User management

A web visualization is closely connected to user management. By configuring a user, you control exactly what the user can view in the web visualization. You must also define corresponding user permissions.

The **admin** user has administrator privileges. This is ideal for engineering because **admin** has access to all elements of the station. For security reasons, this user should not be used for operation of a web visualization. Instead, login credentials with minimal access privileges should be created for each user. This allows recorded user actions to be traced to specific individuals.

User permissions are defined in multiple steps:

6.9.2.1 Restricting access to slots

Slots of a component can have an **Operator** flag that can be defined or deleted during engineering in the **“AX Slot Sheet”** view via the **“Config Flags”** context menu. For points, typically the *Out* slot carries the operator flag. For writable points, the *set* actions for manual control or *override, auto* actions for manual override also carry the operator flag.

All slots with the operator flag require only operator permissions to read, write, and invoke, whereas all other slots have corresponding administrator permissions. The goal is to only assign operator rights to a user of visualization pages and thereby restrict permissions accordingly.

6.9.2.2 Restricting access to components

Components, files, and trend data records can be assigned to categories. Users only receive access permissions for categories that are needed for their task. Typical uses for categories include trades or services.

All components can be individually assigned to categories via the **“Category Sheet”** view. If the Inherit checkbox is selected, the assignment of the higher-level component in the component tree is applied. This view also contains a **“CategoryService”** button that takes you directly to the **“Category Browser”** default view of the Category Service in **“Station / Config / Services / CategoryService”**. The Category Browser allows you to efficiently assign all components, files, and histories to existing categories.

Two categories, *User* and *Admin*, already exist. All components are assigned to the *User* category. You can define additional categories using the **“Category Manager”** view in “Category Service”.

6.9.2.3 User roles

Permissions are defined in detail in user roles to facilitate the configuration of user permissions for many users. The user is then simply assigned one or more of the configured user roles.

User roles are defined in **“RoleService”** under **“Station / Config / Services”** in the **“Role Manager”** default view.

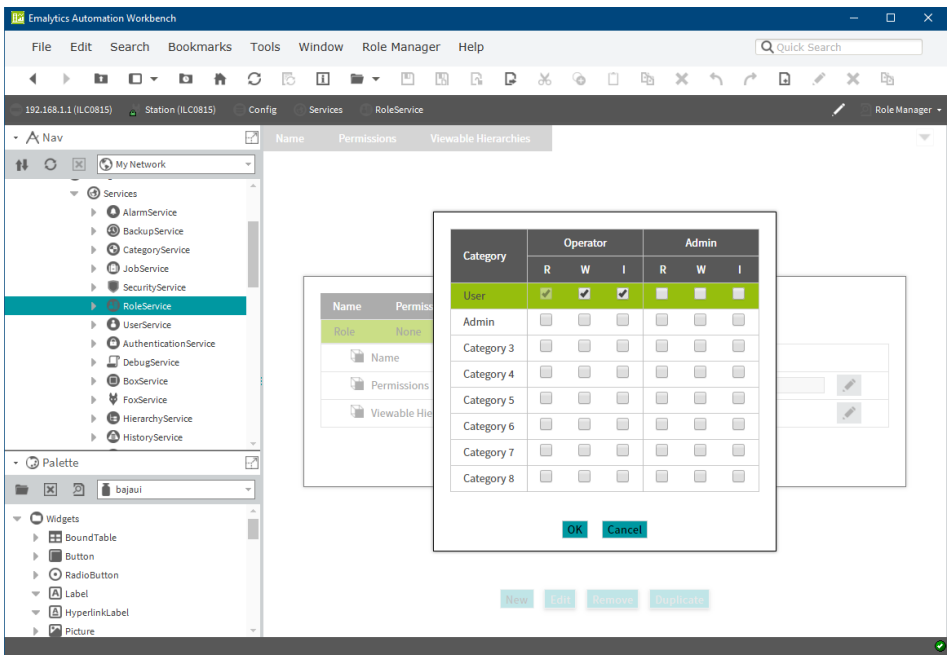


Figure 6-32 Role Manager / user management

- In the “Role Manager”, create a new user role using the **“New”** button and name it *Service*.
- Click the edit icon » in the **“Permissions”** line and in the **“Permissions”** dialog that opens, enter operator rights for read (**R**ead), write (**W**rite), and invoke (**I**nvoke) in the **“User”** category.

6.9.2.4 Users

Users can be created and maintained in the **“AX User Manager”** view of the **“Station / Config / Services / UserService”** user service. Each user has a user name and password for authentication as well as other user-specific properties.

- Use the **“New”** button to create a new user in the **“AX User Manager”**, assign the name *WebUser* and, under **“Authenticator”**, the password *WebVis12345*.
- Under **“Roles”**, select the user role **“Service”**.

6.9.2.5 Navigation

You must now configure the user-defined view for **“MyAdd”** as the home page for the web server. This is done using a **.nav*. navigation file. Create a new *nav* folder in **“Station / Files”** via the local **“New Folder”** menu.

- Create a new nav file in **“Station / Files / nav”** via the local **“New / NavFile.nav”** menu and assign the name *MyAdd.nav*.
- In the default **“Nav File Editor”** view of the created nav file, double-click the **“Home”** line. An **“Edit”** dialog appears. Click the expand arrow on the folder icon in the **“Target Ord”** line to enter the target folder of the **“MyAdd”** folder using the **“Component Chooser”**.

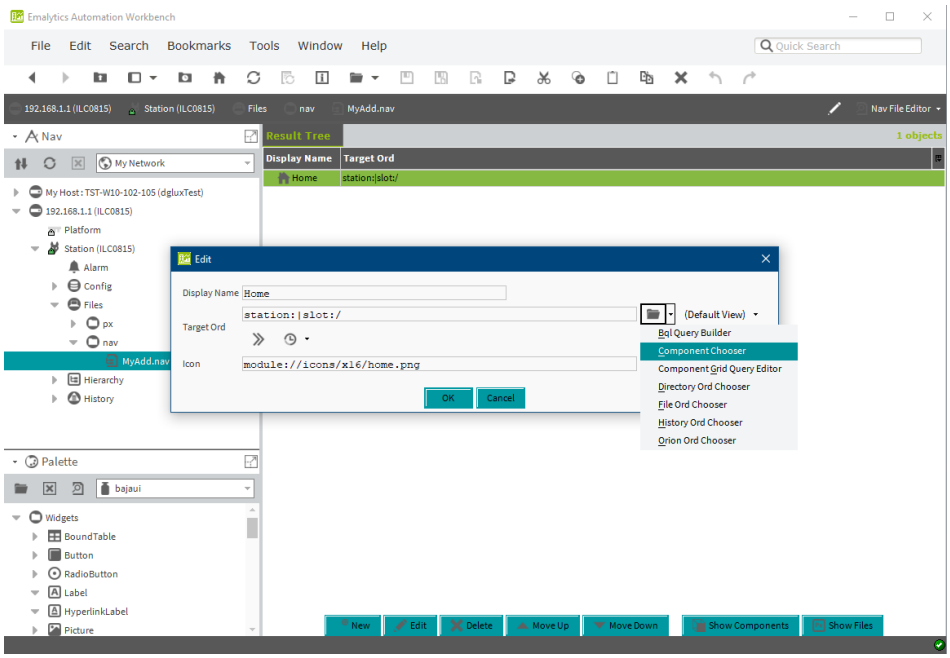


Figure 6-33 Station navigation

In addition to the home page, a basic navigation structure can be added. You must add a **“Home”** branch to the navigation tree. Then, you can configure a suitable navigation structure under this branch.

- To do this, in the **“Nav File Editor”**, drag the following components from the **“Nav”** side bar into the **“Result Tree”** directly below **“Home”**:
  - **“Station / Config / Services / AlarmService / ConsoleRecipient”**
  - **“Station / History / ILC0815 / Add”**
  - **“Station / Config / MyAdd / NumericSchedule”**
- In the Edit dialog of the three entries, enter *Alarm*, *Trend data*, and *Schedule* in the Display Name field.
- In the **“AX User Manager”** for the user **“WebUser”**, select the *nav / MyAdd.nav* nav file under **“Nav File”**.

6.9.2.6 Web profiles

The Emalytics Framework has several defined profiles that determine which functionality is available on the web pages. This functionality ranges from simply display of views contained in the nav file to the web workbench. For each user, a profile can be assigned under **“Default Web Profile”** for contact via the desktop web browser, or under **“Mobile Web Profile”** for contact via browsers of mobile devices.

- For **“Web Users”**, use the **“AX User Manager”** to set all elements except **“Enable Nav Tree Side Bar”** and **“Enable Nav File Tree”** to *No* in the **“Default Web Profile”** of the type *HTML5 Hx Profile*.

6.9.2.7 Https and certificates

The web server of the Emalytics Station can be configured via **“Station / Config / Services / WebService”**. You have a range of options here. Under **“Https Cert”**, you can select the certificate with which the web server will identify itself. Certificates can be created and maintained using the **“Certificate Management”** platform tool.



NOTE

The supplied Tridium certificate can be used for the time being, but it should be replaced with a trusted certificate if secure communication is required.

When you log in to the web server (accessed via https://<ip>:8443), the web browser will display warnings because the Tridium certificate is untrusted.

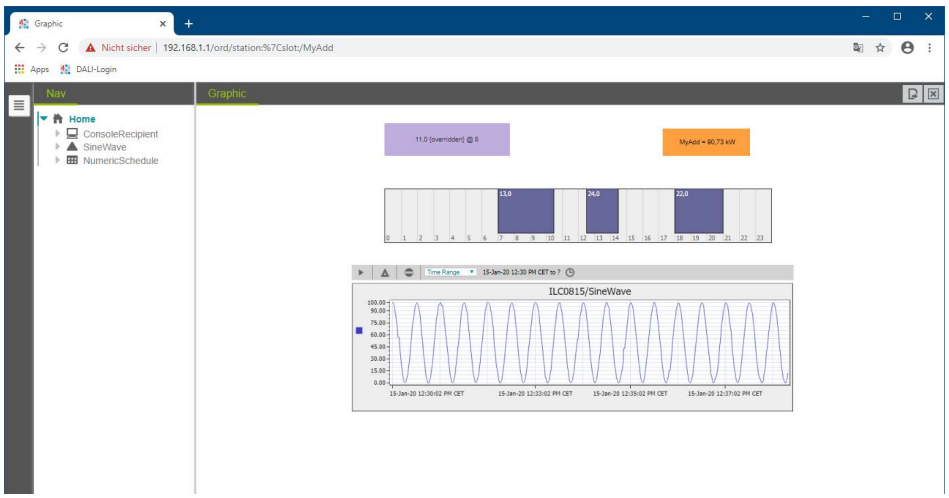


Figure 6-34 Web server warning

6.10 Drivers

Drivers play an essential role in the Emalytics Framework. They are the interfaces to all higher- or lower-level third-party systems. They implement communication protocols, database query languages, or web services and standardize information from third-party systems as Emalytics Automation points.

The Emalytics Automation driver framework makes it very easy for developers to program new drivers for a wide range of applications. There is a large choice of compatible drivers in the Niagara Community: Over 100 drivers are available, with more and more being added all the time.

Below **“Config”**, there is a **“Drivers”** folder containing all communication drivers of the station. Drivers have a uniform structure for the most part. **Devices** are created below the **“Network”** component with protocol-specific properties. These devices have **points**. The user interface, which consists of driver managers, is also standardized. The Network component typically has a default **“Device Manager”**, which helps you create and maintain device components in the station. The **“Points”** folder of a device typically has a **“Point Manager”** as a default view in which points can be created and maintained.

Emalytics Controllers support serial and IP drivers, whereas the Emalytics Supervisor only supports IP drivers. Some drivers can only be used for certain platforms. This includes database drivers that can only be licensed for use with the Emalytics Supervisor, or the SysmikScaIo driver, which is only used with the ILC 2250 BI for controlling the modular I/O of the controller.

Points of Emalytics communication drivers have a driver-specific proxy extension, which forms the link between the communication protocol and the point. This proxy extension is important from a licensing perspective. Each proxy extension requires a point license. These communicative points are called **“proxy points”** to distinguish them from other points.

6.10.1 I/O driver – SysmikScaIoNetwork

The **“sysmikScaIo”** module contains the driver for controlling the modular I/O of the ILC 2250 BI. Open the **“sysmikScaIo-rt”** module via the **“Software Manager”** platform tool. Additional modules may need to be installed due to existing dependencies.

- Double-click **“Station / Config / Drivers”** to open the **“AX Driver Manager”**, where you can easily create drivers in the station.

↪ Click the **“New”** button to display a list of all the drivers included in the Workbench modules.

- Select **“Sysmik Sca Io Network”** to create the network.

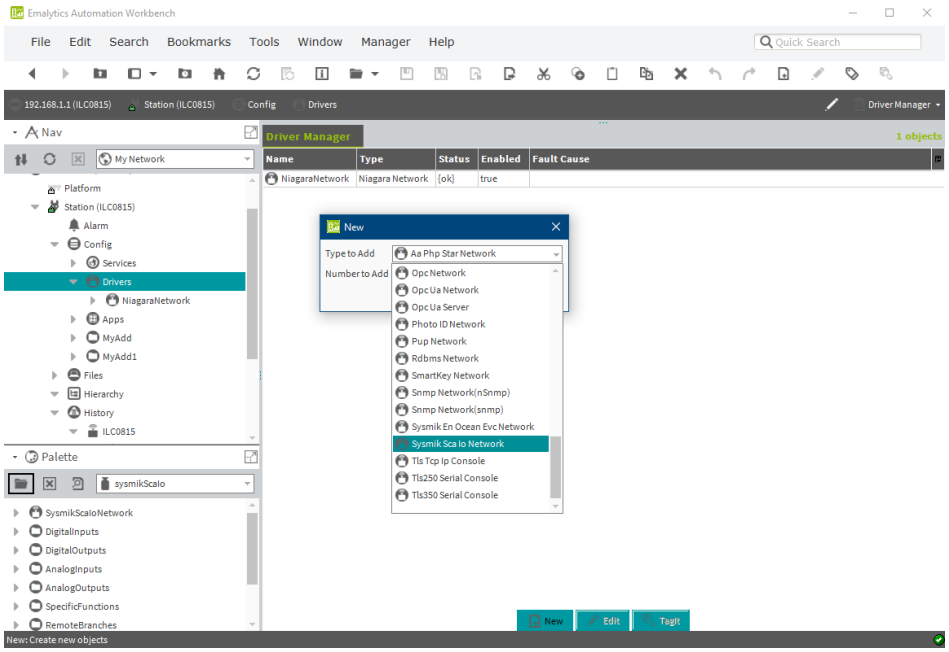


Figure 6-35 Driver management

I/O terminals are controlled largely independently of Niagara by a native I/O driver. This native driver has a network interface for integration with Emalytics. This also makes it possible to control the I/O of one ILC 2250 BI from the station of another ILC 2250 BI. The relevant **“Address”** IP settings of the Niagara driver can be accessed via the **“AX Property Sheet”** of the **“SysmikScaIoNetwork”**.

The native I/O driver can be configured via **“SysmikScaIoNetwork / LocalPlatform”**. In addition to the **“Io Port”**, you can also specify an **“Io Remote Addr”** IP address to permit remote control.
The value **“255.255.255.255”** has a special function here: Remote control is permitted for all IP addresses.

- Double-click the new **“SysmikScaIoNetwork”** to open the **“Sysmik Sca Io Device Manager”**, which helps you create and manage I/O terminals.
- Use the **“Discover”** button to initiate a search. All terminals connected to the controller are detected and displayed in the upper area of the manager.
- Add I/O terminals to the station by selecting them and choosing **“Add”** under **“Config / Drivers / SysmikScaIoNetwork”**.

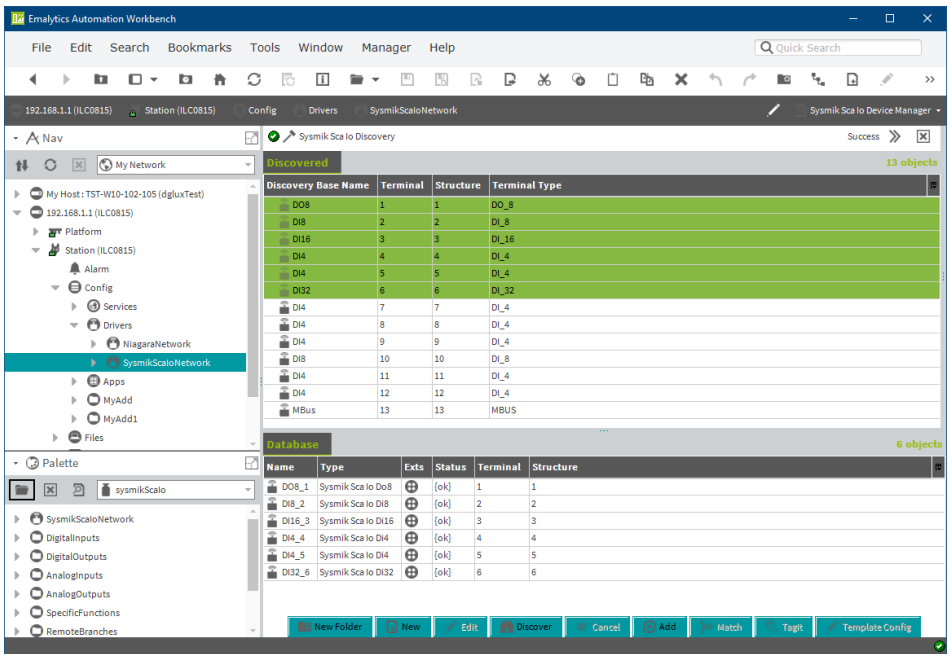


Figure 6-36 Discover search in Device Manager

You can create device folders via **“New Folder”** to add device components below the network component.

Double-click the points icon of an I/O terminal to open the **“Sysmik Sca Io Point Manager”** of the terminal, where you can easily create and manage points. Use the **“Discover”** button to display all I/O points provided by the terminal at the top of the manager. Points can be added to the station by selecting them and choosing **“Add”** under **“Config / Drivers / SysmikScaIoNetwork / <IO Terminal> / Points”**.

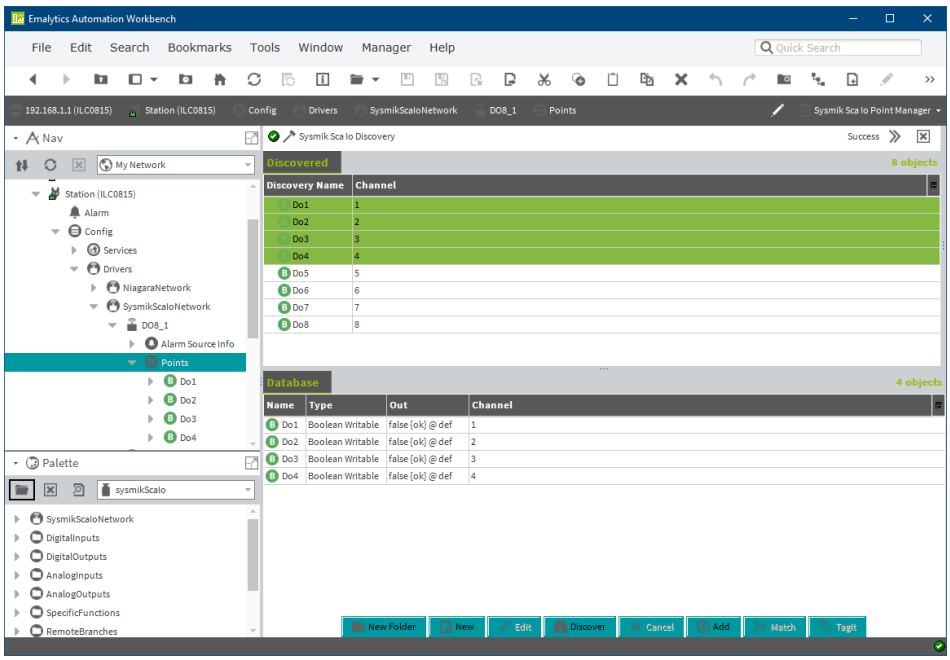


Figure 6-37 Point configuration in the Device Manager

You can create point folders via **“New Folder”** to add points within the “Points” folder.

Communication in the SysmikScaIo driver is event-driven. The driver cyclically communicates with the native I/O driver in the defined poll interval and automatically receives corresponding change messages if the value changes within the poll interval. For each point, one of three poll speeds can be selected via **“ProxyExt / PollFrequency”**. The poll cycle times for all three poll speeds can be set under **“Poll Scheduler”** in the **“AX Property Sheet”** of the **“SysmikScaIoNetwork”**. Longer poll cycle times have no negative effect on the I/O speed due to event-driven communication.

Analog sensors can increase the processor load due to constant, negligible value changes. In this case, the ProxyExtensions of the analog inputs have an UpdateOnDelta property. Here, a relevance threshold can be configured. Changes during a poll cycle below this threshold are ignored.

Many analog input terminals support multiple measuring ranges. These can be configured for each affected point via the **“Sysmik Sca Io Point Manager”** or directly via the *Ai Type* property of the *Proxy Ext* proxy extension. Frequently, electrical variables (voltage, current, resistance) are recorded by sensors that represent completely different physical parameters (pressure, volume flow, temperature).

Conversion rules (linear equation, value table, etc.) for the point can be defined via **“Proxy Ext / Conversion”**. You can assign suitable units to the point via *Facets*.

6.10.2 Serial drivers

Only Emalytics Controllers can communicate serially. The ILC 2250 BI has two serial RS-485 interfaces: COM1 and COM2. Additional serial interfaces can be added as needed using clip-on Inline terminals. Consecutive interface identifiers COM3, COM4, etc. are assigned according to the position of the terminals.

NOTE

Only COM1 and COM2 support the BACnet MS/TP protocol.

IB IL MBUS-type terminals support the M-Bus standard and can only be used with the M-Bus communication driver.

IB IL RS UNI-type terminals support the RS-232, RS-422, and RS-485 interface standards. The interface standard can be selected via the *Rs Type* property of the corresponding **“SysmikScaIoRsUni”** terminal component in the **“SysmikScaIoNetwork”**.

NOTE

Two DIP switches are accessible on the side of the terminal housing of the IB IL RS UNI. These must be positioned as shown in the figure before installing the terminal.

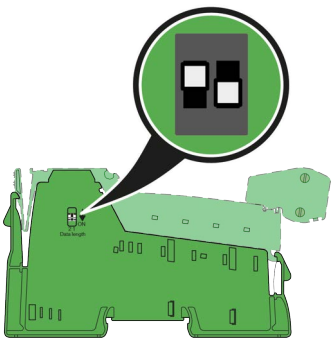


Figure 6-38 IB IL RS UNI – checking the DIP switch position

Example of Modbus/RTU:

- Install the **“modbusAsync-rt”** module using the **“Software Manager”** platform tool.
- Double-click **“Station / Config / Drivers”** to open the **“AX Driver Manager”**. Choose **“New”** to create a new *Modbus Async Network*. The ModbusAsyncNetwork is marked as faulty in the Driver Manager. The specified reason for the error is that no serial port has been assigned: “No port selected for Modbus communication.”
- Open the **“AX Property Sheet”** view and enter interface identifier *COM1* under **“Port Name”** in the **“Serial Port Config”** property. Other parameters such as *Baud Rate* and *Stop Bits* can also be set here.

↪ The status of the network changes to {ok}.

All available serial interfaces with their assigned networks are listed under **“Station / Config / Services / PlatformServices / SerialPortPlatformServiceNpsdk”**.

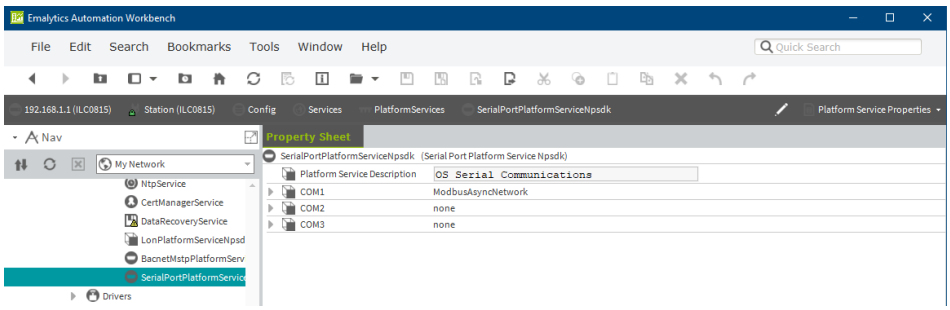


Figure 6-39 Property sheet with overview of available serial interfaces

6.11 Saving a station

6.11.1 Saving a station

A station is run completely from RAM. When you choose **“Station / Run”** or the **“Save”** action in **“Station / Config”**, the station, including trend and alarm data, is stored in the nonvolatile memory of the host. The station is automatically saved when the station is shut down, for example when a module is updated.

In the **“Platform Service Container Plugin”** default view in the **“Station / Config / Services / PlatformServices”** platform service, you can use **“Station Auto-Save Frequency”** to define the interval at which the station will be automatically saved. The default setting is one hour. Emalytics Controllers such as the ILC 2250 BI use flash memory as nonvolatile memory. Flash memory can be damaged by excessive writing activity. Therefore, shortening the interval is not recommended.

The ILC 2250 BI also has an infinitely writable SRAM memory, where all changes that occur during operation are cached. In this way, no data is lost even in the event of a voltage failure.

If necessary, uninterruptible power supplies can be used to protect Supervisor stations against loss of data.

6.11.2 Copying a station

Stations can be copied between Emalytics hosts and the Emalytics Automation Workbench using the **“Station Copier”** platform tool. Trend data and alarms are also transferred in addition to the application. A station copied in this way can run on any Emalytics host. There is no synchronization of version information of module versions or system software. This allows stations to be developed, tested, and commissioned online on an Emalytics Controller or Supervisor, or offline in the Emalytics Workbench file system.

6.11.3 Backup

Use the **“Station / Station Backup”** context menu to back up the station in the file system of the Emalytics Workbench. This is done using a distribution file with the extension .dist.

In addition to the application, trend data, and alarms, references to installed modules and system software are also backed up. A backup can be loaded into suitable hosts with the Distribution File Installer platform tool.

The appropriate modules and system software are automatically installed at the same time. It is strongly recommended that the controller is commissioned beforehand. Make sure that the Niagara versions of the backup and controller match. A typical application is the transfer of software when devices are replaced.

6.12 Application structure

For the most part, the structure of the station under **“Station / Config”** is open. There are some conventions that simplify the maintenance of a station:

– All services are subordinate to the **“Services”** service container
– All drivers or network components of the drivers are subordinate to the **“Drivers”** driver container

Furthermore, many components have restrictions on where they can be placed in the station tree. Typically, the devices of a driver can only be placed below the corresponding network, and the points of a driver can only be placed under the corresponding devices in the **“Points”** folder. Alarm extensions can only be placed below points, and so on.

In addition, the user is free to decide how to structure an application. The best approach in terms of engineering effort over the product lifecycle is extremely dependent on the requirements of the application. For large projects, it is usually necessary to form repetitive structures.

An I/O-based approach is recommended if the control tasks or the structure of the visualization pages follow the structure of the integrated devices. Control algorithms are created directly in the driver structure below the network or devices. One example would be monitoring/visualization of the room automation systems of an office building, with one BACnet room controller per room integrated into Emalytics via a BACnet network.

Often, the application structure cannot be easily mapped to the I/O structure.

For example, you may need to control the lighting control system via DALI, push buttons via Modbus, and room controllers via LON and link them together. In this case, it is best to program the application functions in an I/O-independent structure and use links to connect the application functions and the I/O points.

7 Internal communication within the Emalytics network

The Emalytics network serves for communication within the Emalytics Stations. It allows you to exchange points, trend data, alarms, schedules, and files as well as synchronize login credentials. In addition, you can use provisioning to automate most system maintenance, including updates, upgrades, and backups, from a central location.

NOTE: The station name is used to identify a station in the network. Therefore, you must ensure that station names are unique when planning a station.

7.1 Emalytics Automation Supervisor Station

An Emalytics Station can run on a Windows computer. The Supervisor software must be installed on the computer. This is also referred to as an Emalytics Automation Supervisor Station (or simply Supervisor Station). Supervisor Stations only support IP protocols. Since the performance of the host is potentially much higher, Supervisor Stations often assume a central, higher-level position in the Emalytics system structure and implement the following functions, among others: web visualization, trend data archiving (also including export to external databases, if necessary), and alarm console. Communication with the lower-level stations on Emalytics Controllers takes place via the Emalytics network.

The Emalytics platform of a Supervisor is provided by the “Niagara” Windows service. If this service is not already started, install and start it from the Windows Start menu by choosing **“Emalytics Automation Installation Platform Daemon – 4.<x.x.x>”**.

- Create a new station in the Workbench by choosing **“Tools – New Station”** from the menu.
- Enter the name *Sup* in the **“Station Name”** field and select **“NewSupervisorStation-Windows.ntpl”** as the **“Station Template”**.
- Assign the password *Admin12345* for the **“admin user”**.
- Select the final action, **“copy it to the local platform with Station Copier”**, to send the station to the Emalytics platform daemon.
- You will be prompted to enter your user name and password in order to connect to the platform. Enter your Windows login credentials here.

NOTE: The user requires local admin rights to gain access to the Supervisor platform.

The Emalytics Automation Workbench automatically connects to the platform daemon. The station is transferred to the file system of the platform daemon via the **“Station Copier”** platform tool and started there. You can monitor the startup process via the **“Application Director”** platform tool, which starts automatically.

- Create a station connection from the Workbench to the Supervisor Station by choosing **“File – Open Station”** from the menu.
- Select *TLS WebSocket Connection* for the type and enter *IP localhost* for the host.
- You may be prompted to confirm the public key of the Supervisor. Log in with user name *admin* and password *Admin123345*.

7.2 Communication

The Emalytics network communicates using the **Fox** protocol or the secure **Foxs** protocol based on TLS. These are the same protocols that the Workbench uses to communicate with the station. Communication can be configured via the Fox Service under **“Station / Config / Services / FoxService”**.

By default, only Foxs is enabled on TCP port 4911. The **Foxs Cert** property can be used to select a certificate for communication.



The supplied **Tridium** certificate can be used for the time being, but it should be replaced with a trusted certificate if secure communication is required.

If secure communication is not required, Fox protocol can be enabled via **“Fox Enabled = true”** and **“FoxsOnly = false”**. The communication parameters of the communication partners must match.

Communication is event-driven and is always established from a client to a server. Stations often assume both client and server roles for communication between two stations. This means that each communication partner must be created in the Emalytics network of each station.

7.3 Integrating a station

- Perform these steps in both stations.
- Double-click **“Station / Config / Drivers / NiagaraNetwork”** to open the **“AX Station Manager”** of the Emalytics network. Detection of stations in the network via the **“Discover”** button is based on UDP protocol using multicast. Under certain circumstances, settings must be made to allow for this:

– For multicast communication, a gateway address that matches the IP address must be defined in the IP settings, even if there is no gateway in the network.

Note: If no gateway is used when configuring the network settings, the entry depends on the controller used:

  – **With the ILC 2250 BI(-L), the gateway fields remain empty in this case.**
  – For controllers of the predecessor series ILC 2050 BI(-L), the value "0.0.0.0" must be entered for the gateway.

– Firewalls must be configured to allow UDP communication on the Fox ports (default *1911* for Fox or *4911* for Foxs).

- Manually create a device (Niagara station) in the Emalytics network using the **“New”** button.

7.3.1 Fox

- Enter the name of the Emalytics Station to be integrated, **ILC0815** or **sup**, as well as the IP address of the controller or supervisor PC.
- Make sure that “Use Foxs” is set to *false* if “FoxsOnly = false” has been set in step 7.2.
- Enter user name **admin** and password **Admin 12345**.

Administrator access should not be used in productive systems. Instead, define a network user with appropriately restricted access privileges.

- Use the **“Ping”** action (“Actions / Ping” context menu) of the newly created device on the Emalytics network to verify that communication via Fox is successful. The device status must be {ok}.

7.3.2 Foxs

- Enter the name of the Emalytics Station to be integrated, *ILC0815* or *sup*, as well as the IP address of the controller or supervisor PC. Set the **Fox Port** to *4911* and **Use Foxs** to *true*.
- Enter user name *admin* and password *Admin 12345*.

Administrator access should not be used in productive systems. Instead, define a network user with appropriately restricted access privileges.

A special feature of secure communication via TLS is the use of certificates, which a server uses to authenticate itself. These certificates must be recognized as trusted by the communication partner. Trusted sources of certificates can be maintained in the **“Sys-**

tem Trust Store” tab in the **“Certificate Management”** platform tool. Certificates issued by these sources are automatically trusted. The supplied Tridium certificate does not come from one of these sources and is not trusted for the time being.

The supplied **Tridium** certificate can be used for the time being, but it should be replaced with a trusted certificate if secure communication is required.

Use the **“Ping”** action of the newly created device to check whether communication is possible. The device should assume the status *{down,alarm...}*. In the **“Health / Last Fail Cause”** property of the device, you should see an error message about the failed certificate check: **javax.net.ssl.SSLHandshakeException: java.security.cert.CertificateException: failed certificate validation, failed hostname validation**.

- Open the **“Allowed Hosts”** tab in the **“AX Certificate Management”** platform tool of the client. The tab already contains an entry with the IP address of the server. The red icon indicates that this host is untrusted.
- Select the line and make the host trusted using the **“Approve”** button (icon).

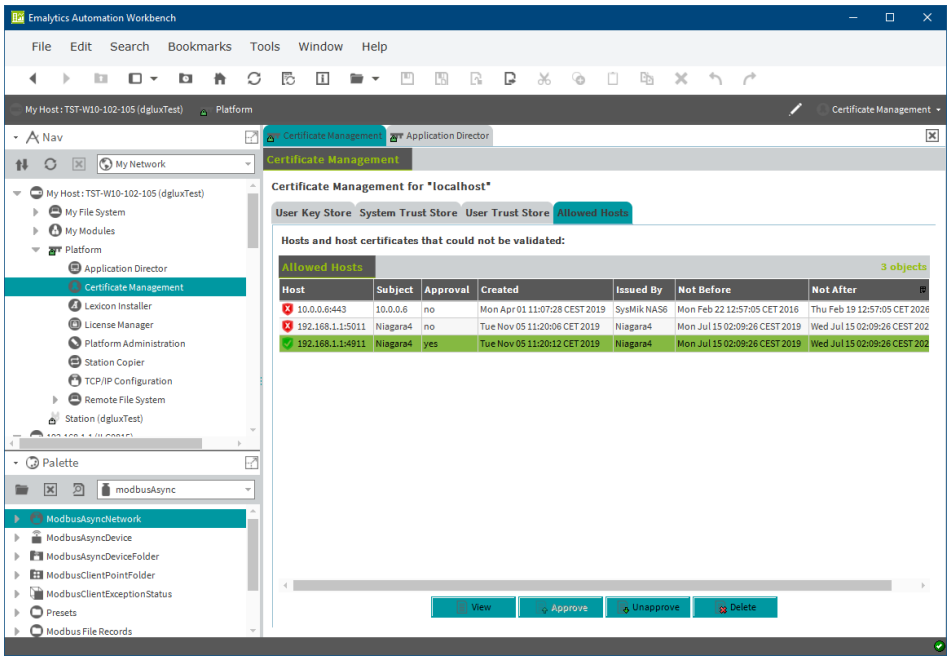


Figure 7-1 Certificate Management

↪ Use the **“Ping”** action of the newly created device on the Niagara network to verify that communication via Foxs is successful. The device status must be *{ok}*, or, until the alarm triggered by the untrusted certificate is acknowledged, it must be *{unackdAlarm}*.

7.4 Integrating points into the Supervisor Station

Every device in the Niagara network has device extensions. Points are integrated via the **“Points”** device extension. Double-click **“Station (sup) / Config / Drivers / NiagaraNetwork / ILC0815 / Points”** to open the **“AX Niagara Point Manager”** default view. Use the **“Discover”** button to search for all **“Control Point”** points in the station of the ILC 2250 BI.

The Emalytics Automation Framework contains a **BQL** query language that can be used to search for information in a station using various search criteria. This query language is used to narrow the search results, but can also be used when creating a report, for example.

Double-click the **“/MyAdd/Add”** point to integrate it into the Supervisor Station. Once integrated, all standard functions of the point, such as alarm management, trend data collection, etc., can now be used in the Supervisor Station.

Points are read-only in the Emalytics network (Niagara network). Due to the reciprocal integration of readable points on both communication partners, both directions of communication are still possible.

7.5 Integrating trend data into the Supervisor Station

Trend data is typically recorded in Emalytics Controllers. The Supervisor Station is better suited for storing data for extended periods because it provides the necessary storage space.

The “Histories” device extension is used for integrating the data recordings of an external station. Double-click **“Station (sup) / Config / Drivers / NiagaraNetwork / ILC0815 / Histories”** to open the **“AX Niagara History Import Manager”**. Use the **“Discover”** button to display all the trend data recordings of the ILC 2250 BI. Double-click the selected, expanded **“Add”** entries or the **“New”** button to add them to the Supervisor Station. This creates a data recording in the Emalytics Supervisor that is cyclically supplied with the data recorded by the ILC 2250 BI. You can define the recording capacity as well as the frequency of data imports. You can also choose *Execute* to initiate the synchronization directly. The data recording can be reached via **“Station (sup) / History / ILC0815 / Add”**.

Trend data can also be exported via the Emalytics network. Communication takes place from the source of the data.

7.6 Integrating schedules into the Supervisor Station

In actual projects, it is common to integrate central schedules of a Supervisor Station into lower-level Emalytics Stations. These schedules can be configured via web visualization.

It also works the other way around: The **“Schedules”** device extension is used for integrating the schedules of an external station.

- Double-click **“Station (sup) / Config / Drivers / NiagaraNetwork / ILC0815 / Schedules”** to open the **“AX Niagara Schedule Import Manager”**.
- Choose the **“Discover”** button to display all the schedules of the ILC 2250 BI.
- Double-click the **“NumericSchedule”** schedule to add it to the Supervisor Station. Imported schedules are read-only.

7.7 Central alarms

In the **“AlarmService”** of the station triggering the alarm (“Station (ILC0815) / Config / Services / AlarmService”), a **“Station Recipient”** from the alarm palette is inserted and connected to the alarm class in order to route alarms of the ILC 2250 BI to the Emalytics Automation Supervisor. The **“Station Recipient”** has a *Remote Station* property in which the *sup* recipient station can be selected.

In the supervisor station, you can use the **“Alarms”** device extension to specify how incoming external alarms will be routed to internal alarm management.

↪ Double-click **“Station (sup) / Config / Drivers / NiagaraNetwork / ILC0815 / Alarms”** to open the **“AX Property Sheet”** view, where you can select the alarm class to which incoming alarms are routed.

8 Security

Niagara systems use IP communication. As a result, it is very important to harden the system against potential attacks. Tridium maintains a **“Niagara 4 Hardening Guide”** for this purpose.

8.1 Passwords

Passwords are used to authenticate users. Never use default passwords. Always use secret, secure passwords that you change on a regular basis. Do not enable the **“Remember These Credentials”** checkbox when you log in to the Emalytics Automation Workbench.

8.2 Access data

Use different login credentials for every user. Delete default login credentials (admin). Only assign permissions that are actually needed.

8.3 Protocols

Use secure protocols (Foxs or https). Create and maintain your own trusted certificates. Disable services or protocols that you do not need.

8.4 Passphrase

A system passphrase is used to encrypt sensitive data of stations such as passwords. Use a secret, secure passphrase.

8.5 Environment

Install Emalytics systems in secure environments. This means installing them in access-controlled rooms or cabinets. This obviously applies to the network infrastructure as well. Protect the Emalytics system against attacks from the Internet by using a VPN.

9 Diagnostics

Whenever a technical system is not behaving as expected, the causes must be determined and eliminated.

9.1 Version information

Prior to contacting Support, version information should be noted and evaluated.

In the **“Platform Administration”** platform tool, you can view all version information via the **“View Details”** button. You can then use the **“Copy to Clipboard”** button to save the information to the clipboard for emailing.

If any software updates or upgrades have been released, they should be installed. Maintenance and support for older versions expires when a new version of Emalytics software is released.

9.2 Event logging

All Emalytics modules can display diagnostic messages. For this purpose, modules have loggers that typically use the same name as the module or a name based on it. Every diagnostic message has a log level to classify the severity of the message from FINEST (detailed information not required for operation) to SEVERE (serious errors). For each logger, a log level can be specified to indicate the minimum severity level at which diagnostic messages will be output via the station output. For most loggers, this level is INFO. This means that all diagnostic messages with the log levels INFO, WARNING, and SEVERE are displayed.

Many modules display highly detailed diagnostic messages with a very low log level. For drivers, these messages may even include protocol records.

9.2.1 Emalytics Station

You can double-click the default view of the **“DebugService”** under **“Station / Config / Services”** to open the **“Logger Configuration”** view. Here, you can define a default level *(ROOT)* for all loggers and deviations from this default level for individual loggers.

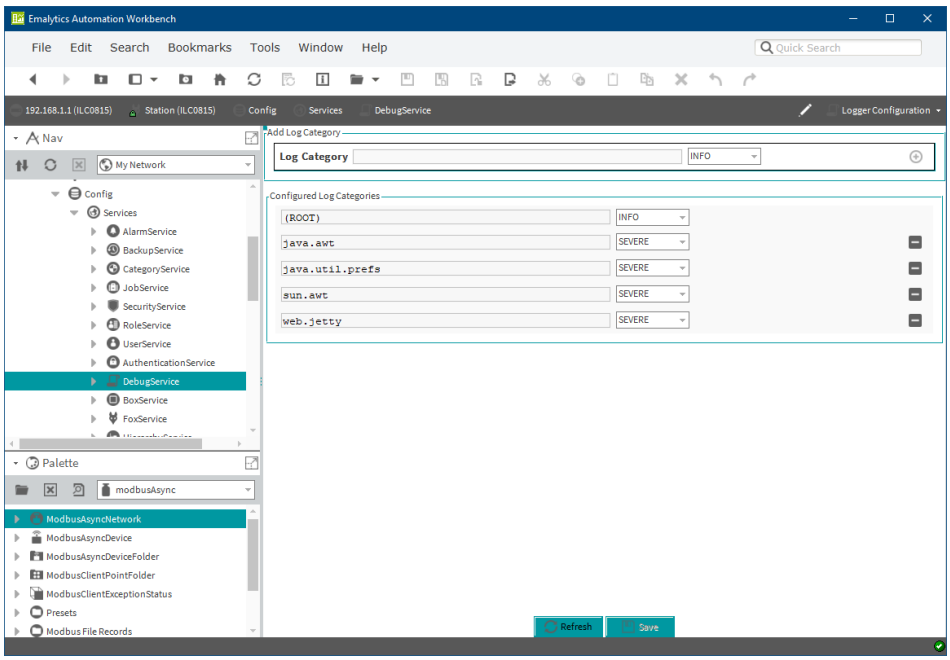


Figure 9-1 Station logger configuration

Messages are output via the station output in the **“Application Director”** platform tool.

Additionally, the **“LogHistoryService”** stores events in the **“LogHistory”** of the station, which can be viewed via **“Station / History / <Station Name> / LogHistory”**.

The **“AuditHistoryService”** logs changes to the station in the **“AuditHistory”**, which can be viewed by choosing **“Station / History / <Station Name> / AuditHistory”**.

9.2.2 Emalytics Automation Workbench

The Emalytics Automation Workbench is used for engineering a station. Components of the user interface such as views, managers, and plug-ins are not executed in the station, but directly in the Workbench. When irregularities appear, it can also be useful to display diagnostic messages for these software components. To do this, start the Emalytics Workbench with the console: **“Workbench N4 (Console)”** start menu.

You can also choose **“Tools – Logger Configuration”** to display the same **“Logger Configuration”** view that is used for the DebugService of the Niagara Station (8.2.1). However, this view is specifically for the Workbench. In other words, you use this view to define the diagnostic messages that the Workbench will display in the external console.

9.3 Execution of applications

The Emalytics Automation Station consists of components whose code is executed by different mechanisms.

Many components that only have processing functions (such as arithmetic operations) calculate their output values only when changes are made to their own inputs.

Components with time functions (e.g., signal generators or PID controllers) are also executed cyclically at intervals that can usually be configured.

Depending on the protocol, outputs of proxy points of the communication drivers are recalculated or read by the **“Poll Scheduler”** when a corresponding protocol frame is received. Only points that are actually required at the moment are read, meaning their values are used for other tasks such as displaying an active view, issuing alarms, recording trend data, or performing calculations. Writing, on the other hand, is typically event-driven and is implemented when a value changes.

All events are decoupled by queues. Event-driven processing is highly effective. However, there is no fixed cycle time and no reliable information on system responsiveness.

As a guideline, the processor load (**“Resource Manager”** view of the station) should not exceed 80%. The processor load can be reduced by extending execution cycles/poll cycles and reducing the number of value changes.

9.4 System diagnostics – Spy

The Emalytics Framework includes system diagnostics functions that are consolidated in the **“Spy”** context menu of the connected station. Only a few of these functions will be mentioned here.

9.4.1 EngineManager

You can obtain information about the processing of the application by choosing **“sysManagers – engineManager”**. When processor load is high, you can choose **“Engine Hogs”** to identify components that are taking up a disproportionately large amount of computing time.

9.4.2 ClassLoaders

Choose **“classLoaders”** to display a list of all installed modules. This list also indicates the number of installed elements of these modules as well as the amount of memory they are using. Modules consuming a disproportionately large amount of memory can be easily identified here.

9.4.3 Log setup

Choose **“logSetup”** to display an alternative view for setting up the logger configuration (8.2.1).

10 Software maintenance

All components of the Emalytics Framework are versioned. This includes the system software as well as all Emalytics Automation modules. Updates within a minor version as well as upgrades to a higher minor version are implemented by reinstalling Emalytics via “Emalytics Setup”. Experienced users can also perform separate installations of Emalytics Supervisor and Emalytics add-ons via a user-specific setup.

Note the following with regard to licensing for upgrades: When a new minor version is released, licenses are only automatically applied to the new version for the duration of the software maintenance period.

10.1 Emalytics Automation Workbench

When Emalytics Supervisor is installed, the Workbench version is part of the proposed installation path. In this way, multiple versions of the Emalytics Automation Workbench can be operated independently on a single PC.

Emalytics add-ons are always related to an Emalytics version and can only be installed in the installation directories of corresponding Supervisor versions.

10.2 Emalytics Automation Supervisor

Software for the Supervisor is not installed via a commissioning process in the Workbench, but directly via the Supervisor installation. A Supervisor and Workbench with the same version use the same Emalytics Automation modules.

Only one platform daemon can be active on a computer. The installation of this Windows service can be activated during the Supervisor installation or later via the **“Platform Daemon Installation”** start menu. The installation will automatically stop and uninstall an already active platform daemon from another Emalytics version.

When the Emalytics software is updated or upgraded, the Supervisor Station can be copied back from the Supervisor to the Workbench using the **“Station Copier”** platform tool, and after the new platform daemon is installed, it can be copied back from the Workbench to the new Supervisor.

10.3 IB IL 2050 BI Emalytics Controller

Updates and upgrades should always be installed through a commissioning process. Individual updates to modules can also be installed using the **“Software Manager”** platform tool.

If you want to downgrade the system software, install the appropriate **clean distribution** for the Emalytics host via the **“Cleaning”** button in the **“Distribution File Installer”** platform tool. This is a minimal installation from which an Emalytics version can be installed using the Commissioning Wizard.

A Technical appendix

B Appendixes

B 1 List of figures

Section 1

Section 2

Figure 2-1: Station engineering in Emalytics 14
Figure 2-2: Example of Emalytics system structure 15

Section 3

Figure 3-1: Workbench start screen 19

Section 4

Figure 4-1: Commissioning the hardware 21
Figure 4-2: ILC 2250 BI web-based management home page 22

Section 5

Figure 5-1: Starting the New Station Wizard 24
Figure 5-2: New Station Wizard administration 24
Figure 5-3: Platform connection establishment 25
Figure 5-4: Platform SSL connection establishment 25
Figure 5-5: Change Platform Defaults Wizard 26
Figure 5-6: Certification confirmation 26
Figure 5-7: Parameterizing a passphrase 27
Figure 5-8: Parameterizing platform access data 27
Figure 5-9: Summary of the changes 28
Figure 5-10: Commissioning: Platform tools 28
Figure 5-11: Commissioning checklist 29
Figure 5-12: Commissioning Wizard: License installation 30
Figure 5-13: Commissioning Wizard: Available licenses are installed via the “Add” button. 30
Figure 5-14: Runtime profile 31
Figure 5-15: Station installation 32
Figure 5-16: Installing software modules 33

Figure 5-17: Installing the system software 34
Figure 5-18: IP configuration 35
Figure 5-19: User configuration 36
Figure 5-20: Completing commissioning 37

Section 6

Figure 6-1: Station overview 41
Figure 6-2: Software Manager workbench 42
Figure 6-3: Workbench workspace 43
Figure 6-4: AX Property Sheet 44
Figure 6-5: Connecting components 45
Figure 6-6: Point palette 46
Figure 6-7: Manual/Emergency/Automatic mode menu structure 47
Figure 6-8: Writable points 48
Figure 6-9: Property facet 48
Figure 6-10: Navigation bar 50
Figure 6-11: History extension manager 51
Figure 6-12: Trend data collection 52
Figure 6-13: Wire Sheet – alarm console 53
Figure 6-14: Property sheet for Out Of Range alarm 54
Figure 6-15: Acknowledge alarms 54
Figure 6-16: Calendar / Schedule workbench 56
Figure 6-17: Creating new views 57
Figure 6-18: Px Editor with side bar 58
Figure 6-19: Adding a widget 59
Figure 6-20: Configuring a widget 59
Figure 6-21: Widget binding 60
Figure 6-22: Defining widget properties 61
Figure 6-23: Modified widget 61
Figure 6-24: Animated BF string 61
Figure 6-25: Absolute Ords 62
Figure 6-26: Relative Ords 62
Figure 6-27: Widget with wizard 63
Figure 6-28: Widget wizard – slider 64
Figure 6-29: Visualized calendar function 64
Figure 6-30: Widget with history chart visualization (time range) 64
Figure 6-31: HVAC kit graphic (example) 65

Figure 6-32: Role Manager / user management 66
Figure 6-33: Station navigation 67
Figure 6-34: Web server warning 69
Figure 6-35: Driver management 70
Figure 6-36: Discover search in Device Manager 71
Figure 6-37: Point configuration in the Device Manager 72
Figure 6-38: IB IL RS UNI – checking the DIP switch position 73
Figure 6-39: Property sheet with overview of available serial interfaces 74

Section 7

Figure 7-1: Certificate Management 81

Section 8

Section 9

Figure 9-1: Station logger configuration 86

Section 10

Appendix A

Appendix B

B 2 List of tables

Section 1

Section 2

Section 3

Section 4

Section 5

Table 5-1: Process status LEDs (marking) 38
Table 5-2: Process status LEDs (function) 38

Section 6

Section 7

Section 8

Section 9

Section 10

Appendix A

Appendix B

B 3 Index

D

Drivers 69

E

Emalytics 13, 16
- Add-ons 17
- View 17

Emalytics Automation 13
- Framework 13
- Frameworks 13, 14
- Host 13
- Modules 14
- Platform 13
- Station 14, 15
- Supervisor 13, 17
- Supervisor Station 78
- Workbench 13, 14, 16

Emalytics building management system (BMS) 16

Emalytics Controller 13, 15

Emalytics modules

I

ILC 2050 BI 13, 15, 21

M

Modules, see Emalytics Automation
- Modules

N

Niagara 13

P

Platform daemon 14

S

Station, see Emalytics Automation Station

Supervisor Station, see Emalytics Automation Supervisor Station

U

Unauthorized network access 11

W

Workbench 16, 17

Please observe the following notes

General Terms and Conditions of Use for technical documentation

Phoenix Contact reserves the right to alter, correct, and/or improve the technical documentation and the products described in the technical documentation at its own discretion and without giving prior notice, insofar as this is reasonable for the user. The same applies to any changes that serve the purpose of technical progress.

The receipt of technical documentation (in particular user documentation) does not constitute any further duty on the part of Phoenix Contact to furnish information on modifications to products and/or technical documentation. You are responsible to verify the suitability and intended use of the products in your specific application, in particular with regard to observing the applicable standards and regulations. All information made available in the documentation data is supplied without any accompanying guarantee, whether expressly mentioned, implied or tacitly assumed.

In general, the provisions of the current General Terms and Conditions of Phoenix Contact apply exclusively, in particular as concerns any warranty liability.

This manual, including all illustrations contained herein, is copyright protected. Any changes to the contents or the publication of extracts of this document are prohibited.

Phoenix Contact reserves the right to register its own intellectual property rights for the product identifications of Phoenix Contact products that are used here. Registration of such intellectual property rights by third parties is prohibited.

Other product identifications may be afforded legal protection, even where they may not be indicated as such.

How to contact us

Internet

Up-to-date information on Phoenix Contact products and our Terms and Conditions can be found on the Internet at:
phoenixcontact.com

Make sure you always use the latest documentation.
It can be downloaded at:
phoenixcontact.com/products

Subsidiaries

If there are any problems that cannot be solved using the documentation, please contact your Phoenix Contact subsidiary.
Subsidiary contact information is available at phoenixcontact.com.

Published by

Phoenix Contact GmbH & Co. KG
Flachsmarktstraße 8
32825 Blomberg
GERMANY

Phoenix Contact Development and Manufacturing, Inc.
586 Fulling Mill Road
Middletown, PA 17057
USA

Should you have any suggestions or recommendations for improvement of the contents and layout of our manuals, please send your comments to:
tecdoc@phoenixcontact.com

Phoenix Contact GmbH & Co. KG
Flachsmarktstraße 8
32825 Blomberg, Germany
Phone: +49 5235 3-00
Fax: +49 5235 3-41200
Email: info@phoenixcontact.com
phoenixcontact.com



© Phoenix Contact 2026-06-10

112333_en_01