

Belimo Digital Ecosystem

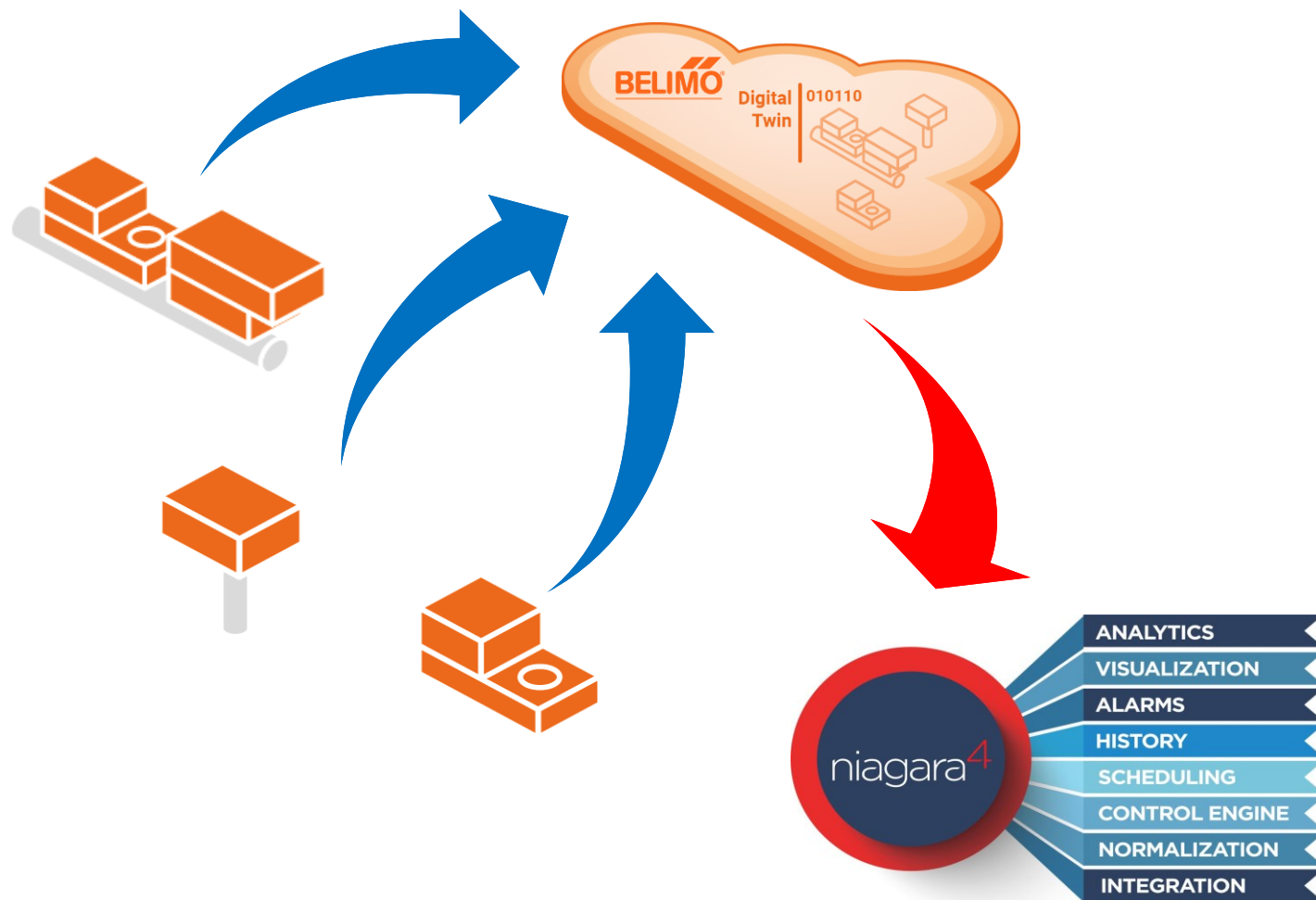
The Niagara Driver for connecting to the Belimo Cloud API

Contents

This is a guided demonstration about installation, activation and use of Belimo driver for Niagara framework.

The driver

The Belimo driver for Niagara framework allows N4 supervisor or Jace 8000 to display your Belimo Energy valve data.



Requirements

- Basic knowledge of Niagara 4 framework
- A Workbench PC installation
- A supervisor licensed installation
- Enough licensed resource points
- An active account on the Belimo Cloud

Installation

Next steps are about installation
of the driver before its use

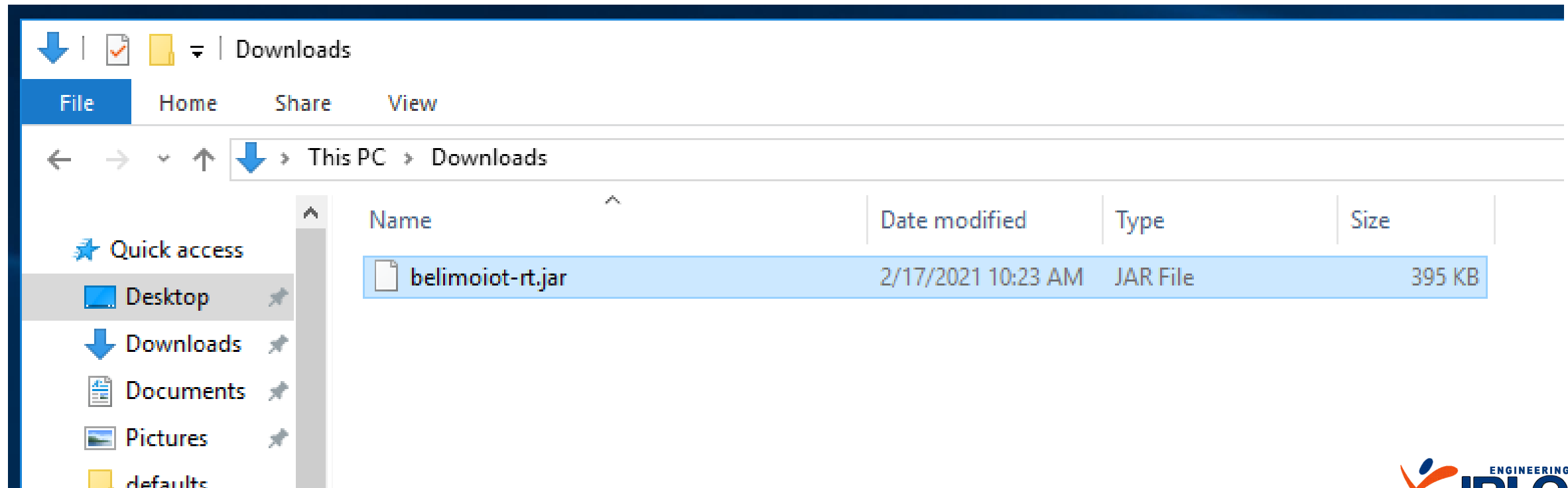
SCAN ME



The jar module

The driver is distributed by INLON for free in this first version. The module and documentation can be downloaded from

github.com/inlon-engineering/niagara-belimo-iot



Copy into Workbench PC

When receiving updated or new module jar files, you have to copy them to your Workbench PC, as follows.

- Close the Workbench application.
- Copy directly into your !/modules directory. This makes the module(s) available in your Workbench environment, and also available to install in other remote platforms (when the installer runs, the module(s) are also copied into your software database, available for installation).

The Niagara modules folder

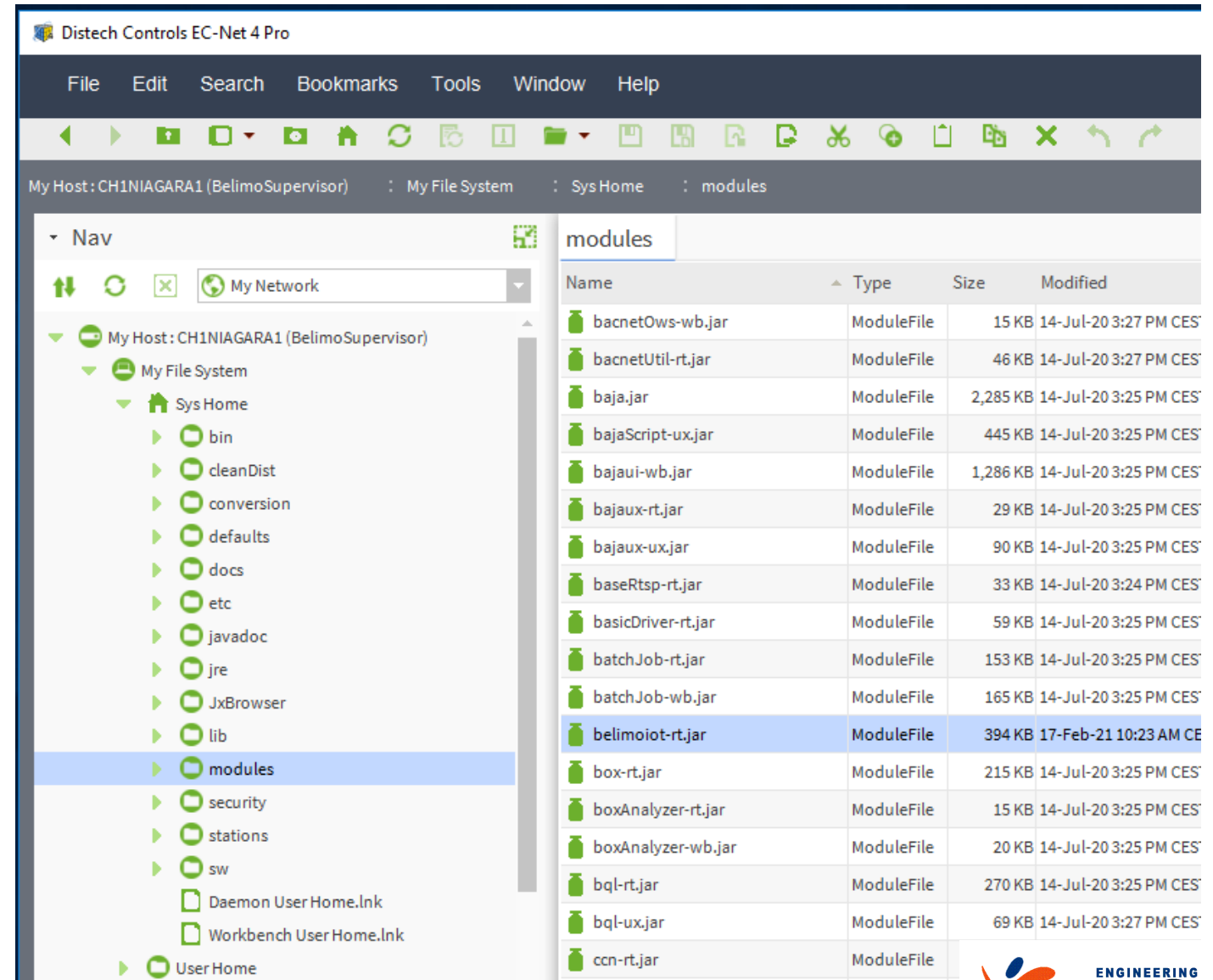
Take care to copy the module into the right current running version of Niagara platform.

The screenshot shows a Windows File Explorer window with the address bar set to 'This PC > Local Disk (C:) > Niagara > EC-Net4-4.9.0.60 > modules'. The left sidebar shows the 'Local Disk (C:)' selected. The main pane displays a list of files with columns for Name, Date modified, Type, and Size. The file 'belimoiot-rt.jar' is highlighted.

Name	Date modified	Type	Size
batchJob-rt.jar	7/14/2020 3:25 PM	JAR File	154 KB
batchJob-wb.jar	7/14/2020 3:25 PM	JAR File	166 KB
belimoiot-rt.jar	2/17/2021 10:23 AM	JAR File	395 KB
boxAnalyzer-rt.jar	7/14/2020 3:25 PM	JAR File	16 KB
boxAnalyzer-wb.jar	7/14/2020 3:25 PM	JAR File	21 KB
box-rt.jar	7/14/2020 3:25 PM	JAR File	216 KB
bql-rt.jar	7/14/2020 3:25 PM	JAR File	27 KB

Check the presence of module

Open the Workbench application and search for belimoiot-rt module into system module folder.



The screenshot displays the Distech Controls EC-Net 4 Pro application window. The interface includes a menu bar (File, Edit, Search, Bookmarks, Tools, Window, Help) and a toolbar with various icons. The main area is divided into a navigation pane on the left and a file list pane on the right.

Navigation Pane (Left):

- My Host: CH1NIAGARA1 (BelimoSupervisor)
 - My File System
 - Sys Home
 - bin
 - cleanDist
 - conversion
 - defaults
 - docs
 - etc
 - javadoc
 - jre
 - JxBrowser
 - lib
 - modules** (highlighted)
 - security
 - stations
 - sw
 - Daemon User Home.Ink
 - Workbench User Home.Ink
 - User Home

File List Pane (Right):

The 'modules' folder is selected, showing a list of files with columns: Name, Type, Size, and Modified.

Name	Type	Size	Modified
bacnetOws-wb.jar	ModuleFile	15 KB	14-Jul-20 3:27 PM CES'
bacnetUtil-rt.jar	ModuleFile	46 KB	14-Jul-20 3:27 PM CES'
baja.jar	ModuleFile	2,285 KB	14-Jul-20 3:25 PM CES'
bajaScript-ux.jar	ModuleFile	445 KB	14-Jul-20 3:25 PM CES'
bajaii-wb.jar	ModuleFile	1,286 KB	14-Jul-20 3:25 PM CES'
bajaux-rt.jar	ModuleFile	29 KB	14-Jul-20 3:25 PM CES'
bajaux-ux.jar	ModuleFile	90 KB	14-Jul-20 3:25 PM CES'
baseRtsp-rt.jar	ModuleFile	33 KB	14-Jul-20 3:24 PM CES'
basicDriver-rt.jar	ModuleFile	59 KB	14-Jul-20 3:25 PM CES'
batchJob-rt.jar	ModuleFile	153 KB	14-Jul-20 3:25 PM CES'
batchJob-wb.jar	ModuleFile	165 KB	14-Jul-20 3:25 PM CES'
belimoiot-rt.jar	ModuleFile	394 KB	17-Feb-21 10:23 AM CES'
box-rt.jar	ModuleFile	215 KB	14-Jul-20 3:25 PM CES'
boxAnalyzer-rt.jar	ModuleFile	15 KB	14-Jul-20 3:25 PM CES'
boxAnalyzer-wb.jar	ModuleFile	20 KB	14-Jul-20 3:25 PM CES'
bql-rt.jar	ModuleFile	270 KB	14-Jul-20 3:25 PM CES'
bql-ux.jar	ModuleFile	69 KB	14-Jul-20 3:27 PM CES'
ccn-rt.jar	ModuleFile		

Module signing advice

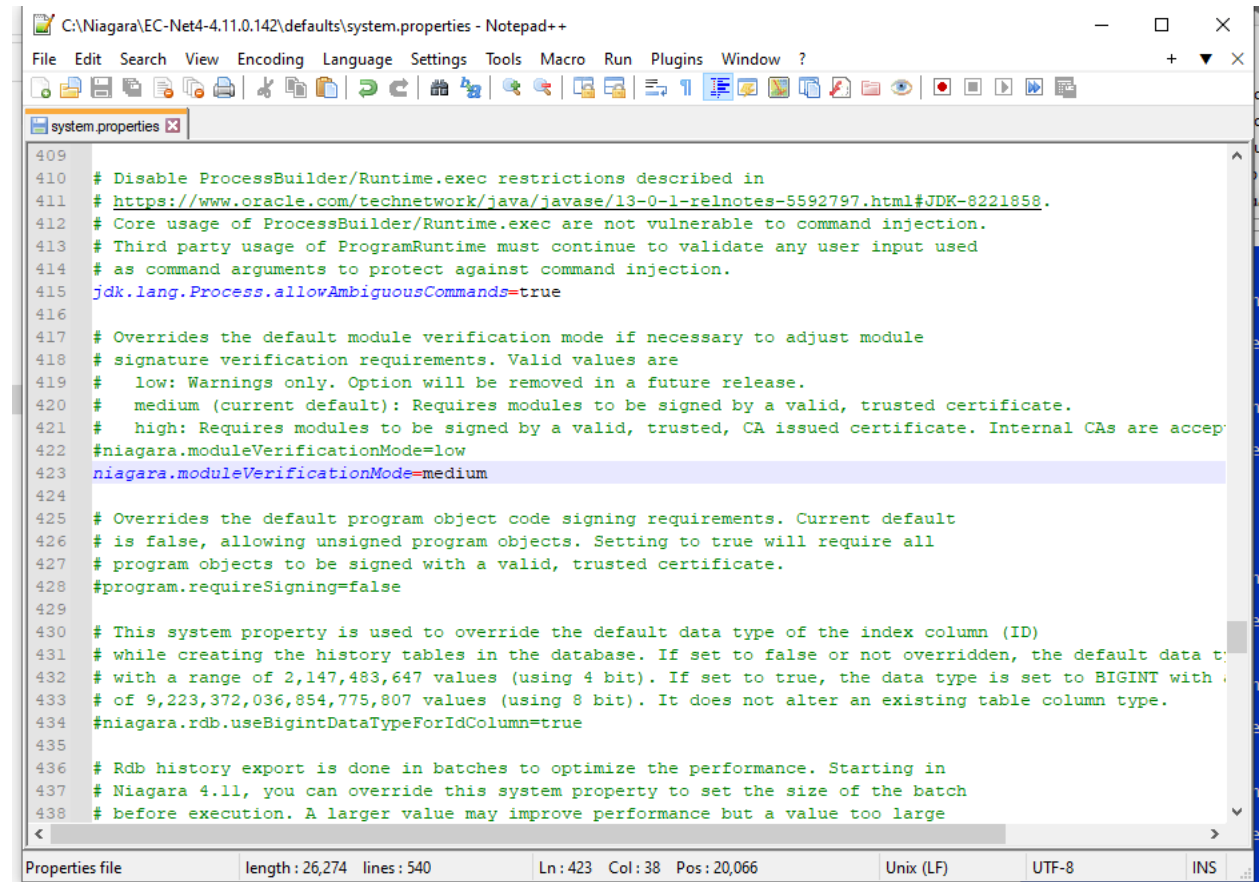
- In Niagara 4.8 and later, there is added support for the signing and verification of third-party modules. Third-party module signing is still optional in most cases, but this will gradually shift to a requirement over the course of the next releases.
- In Niagara 4.8 and later, there are added Workbench tools to support module signing.
- The Jar Signer Tool is useful for non-developers or anyone using an unsigned legacy module. The Jar Signer allows you to “sign” an unsigned *.jar file using a code signing certificate.

Please you can read about module signing here:

<https://docs.Niagara-community.com/bundle/ModuleSigning/page/ExampleConcept-41FE0D3F.html>

Check Workbench module signing level

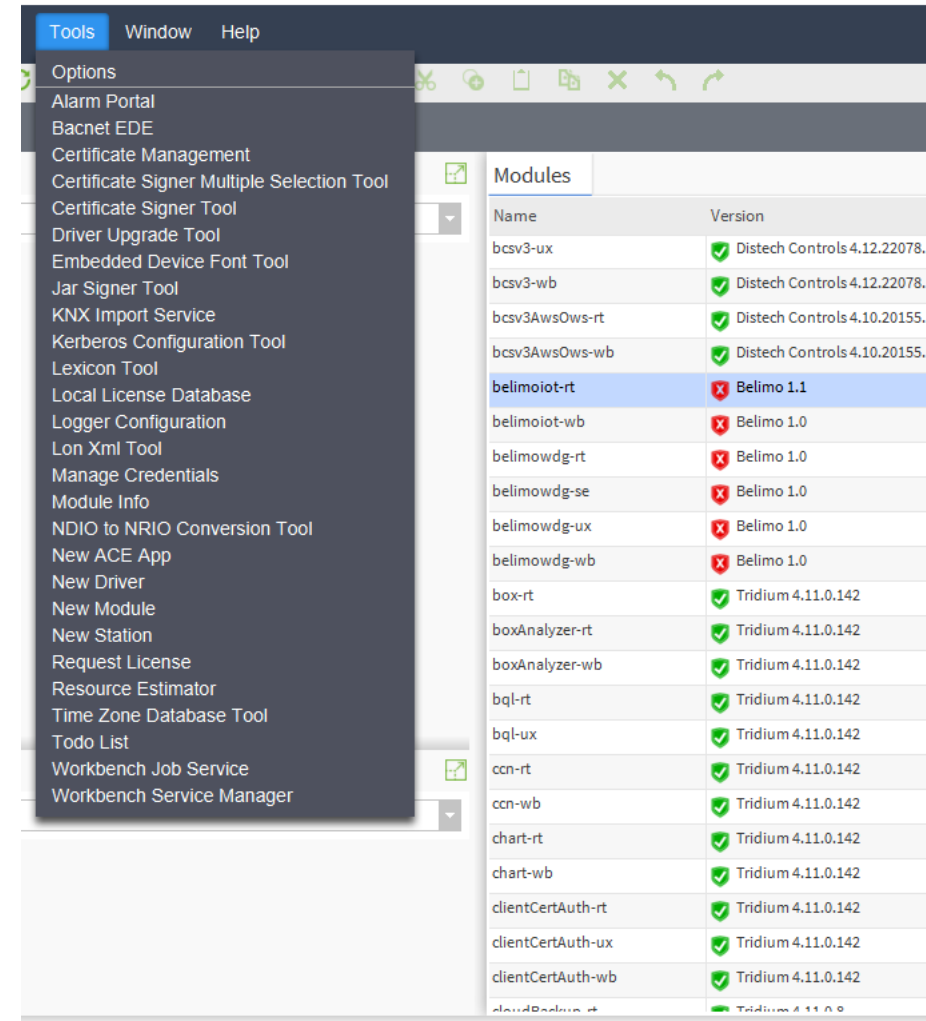
Open the system.properties file of Niagara current version and check what module verification mode is set. Low option is used for development and debugging purposes. Medium level requires external module to be signed.



```
C:\Niagara\EC-Net4-4.11.0.142\defaults\system.properties - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
system.properties
409
410 # Disable ProcessBuilder/Runtime.exec restrictions described in
411 # https://www.oracle.com/technetwork/java/javase/13-0-1-relnotes-5592797.html#JDK-8221858.
412 # Core usage of ProcessBuilder/Runtime.exec are not vulnerable to command injection.
413 # Third party usage of ProgramRuntime must continue to validate any user input used
414 # as command arguments to protect against command injection.
415 jdk.lang.Process.allowAmbiguousCommands=true
416
417 # Overrides the default module verification mode if necessary to adjust module
418 # signature verification requirements. Valid values are
419 #   low: Warnings only. Option will be removed in a future release.
420 #   medium (current default): Requires modules to be signed by a valid, trusted certificate.
421 #   high: Requires modules to be signed by a valid, trusted, CA issued certificate. Internal CAs are accept
422 #niagara.moduleVerificationMode=low
423 niagara.moduleVerificationMode=medium
424
425 # Overrides the default program object code signing requirements. Current default
426 # is false, allowing unsigned program objects. Setting to true will require all
427 # program objects to be signed with a valid, trusted certificate.
428 #program.requireSigning=false
429
430 # This system property is used to override the default data type of the index column (ID)
431 # while creating the history tables in the database. If set to false or not overridden, the default data t
432 # with a range of 2,147,483,647 values (using 4 bit). If set to true, the data type is set to BIGINT with
433 # of 9,223,372,036,854,775,807 values (using 8 bit). It does not alter an existing table column type.
434 #niagara.rdb.useBigintDataTypeForIdColumn=true
435
436 # Rdb history export is done in batches to optimize the performance. Starting in
437 # Niagara 4.11, you can override this system property to set the size of the batch
438 # before execution. A larger value may improve performance but a value too large
439
440
Properties file    length : 26,274  lines : 540    Ln : 423  Col : 38  Pos : 20,066    Unix (LF)    UTF-8    INS
```

Check module sign

Open the module info tool and according to sign status of belimoiot-rt perform the next operations.



Module sign

If red sign is displayed, module cannot be used. The sign operation requires the to import the certificate locally. Ask for “.pem” file to INLON support.



There are errors that are not acceptable for the current module verification mode.



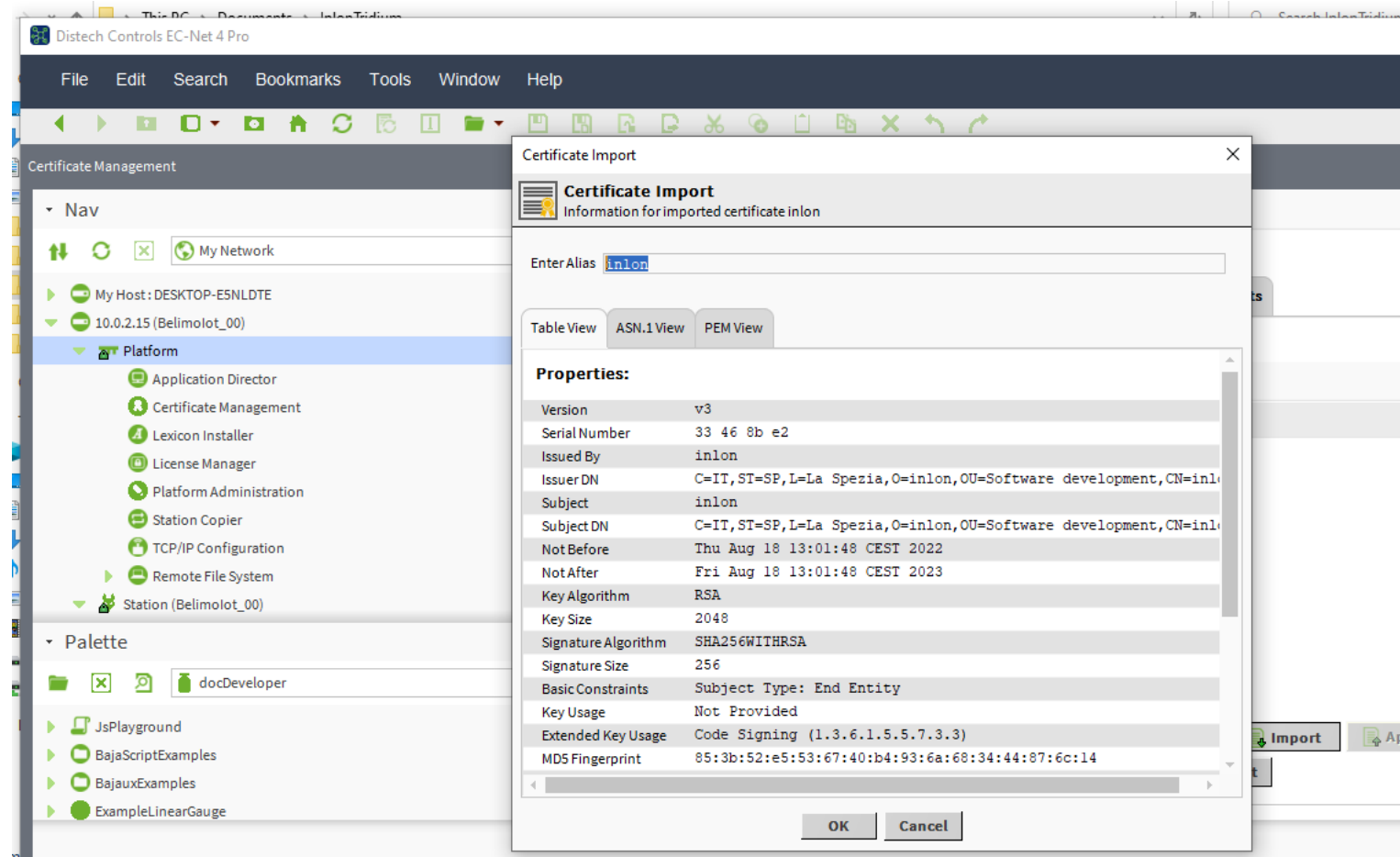
There are warnings, but they are acceptable for the current module verification mode.



Indicates that the module is signed with a Certificate Authority (CA).

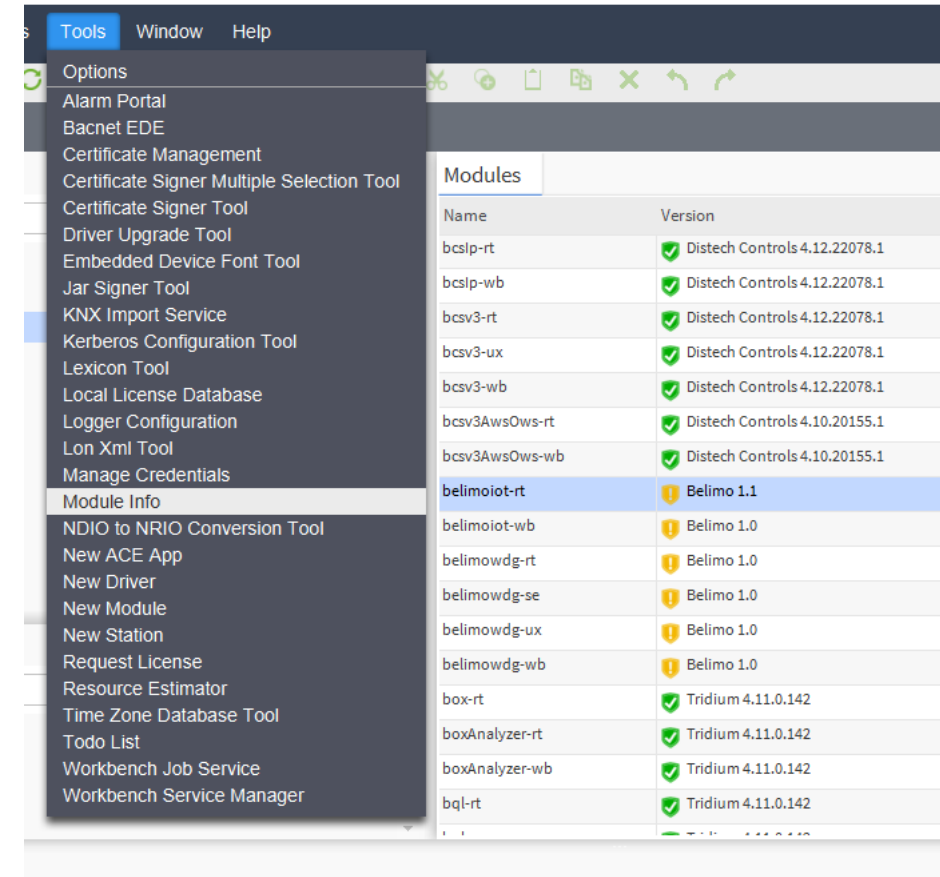
Import certificate to Workbench

Confirm the certificate data clicking ok button.



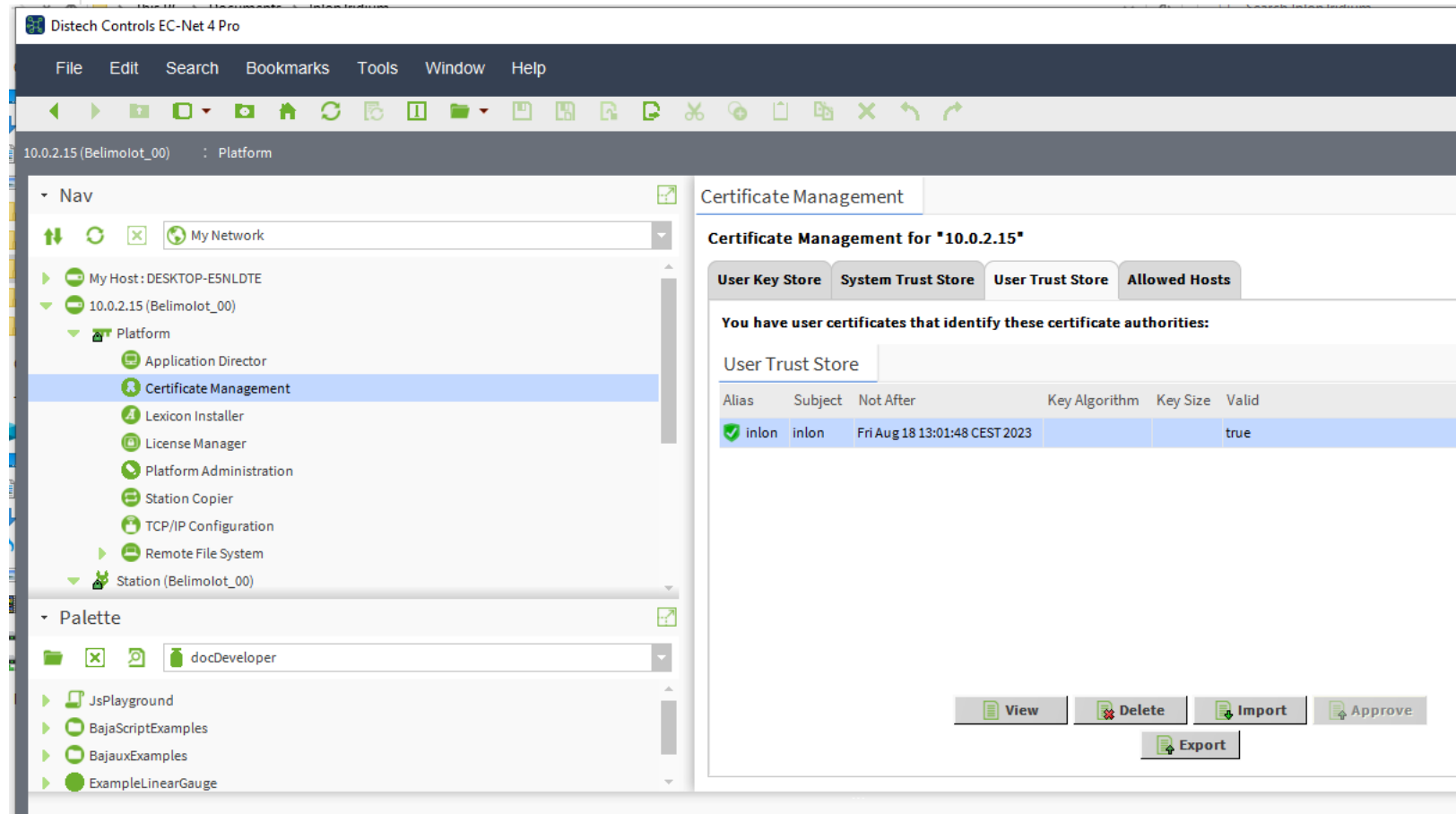
Check module sign again

Open the module info tool and now the belimoiot-rt is signed and can be used.



Import certificate to Platform

The certificate should be installed on stations too. Select the Platform station Certificate Management and perform same operation like for workbench. The Niagara Help describes deeply all the operations.



Using the module

Next steps require
a running station

Load the palette

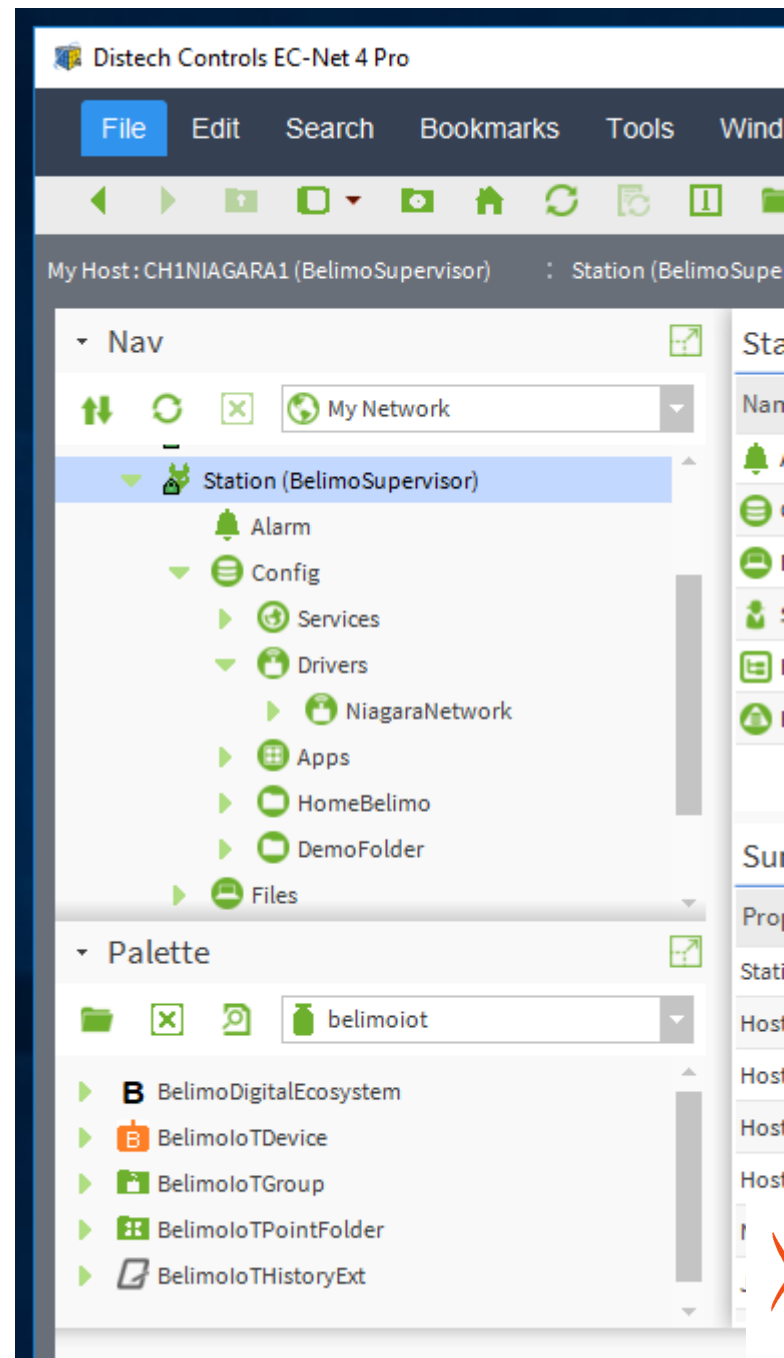
Open the palette view and select the belimoiot one.

The screenshot shows the INLON software interface. On the left, the 'Nav' pane shows a tree structure with 'Station (BelimoSupervisor)' selected. Below it, the 'Palette' pane shows 'kitControl' and 'ControlPalette'. The 'Open Palette' dialog box is open, displaying a list of modules. The 'belimoiot' module is highlighted at the bottom of the list.

Module	Description
bacnetAlarmRouter	Allows re-assignment of alarm classes for Bacnet recv. alarms
bacnetAws	Niagara BACnet AWS Driver
bacnetOws	Niagara BACnet OWS Driver
bacnetUtil	Niagara BACnet Utilities
baja	Niagara Framework
bajaui	Niagara User Interface Framework
batchJob	Batch Jobs
belimoiot	Belimo driver to connect cloud resources

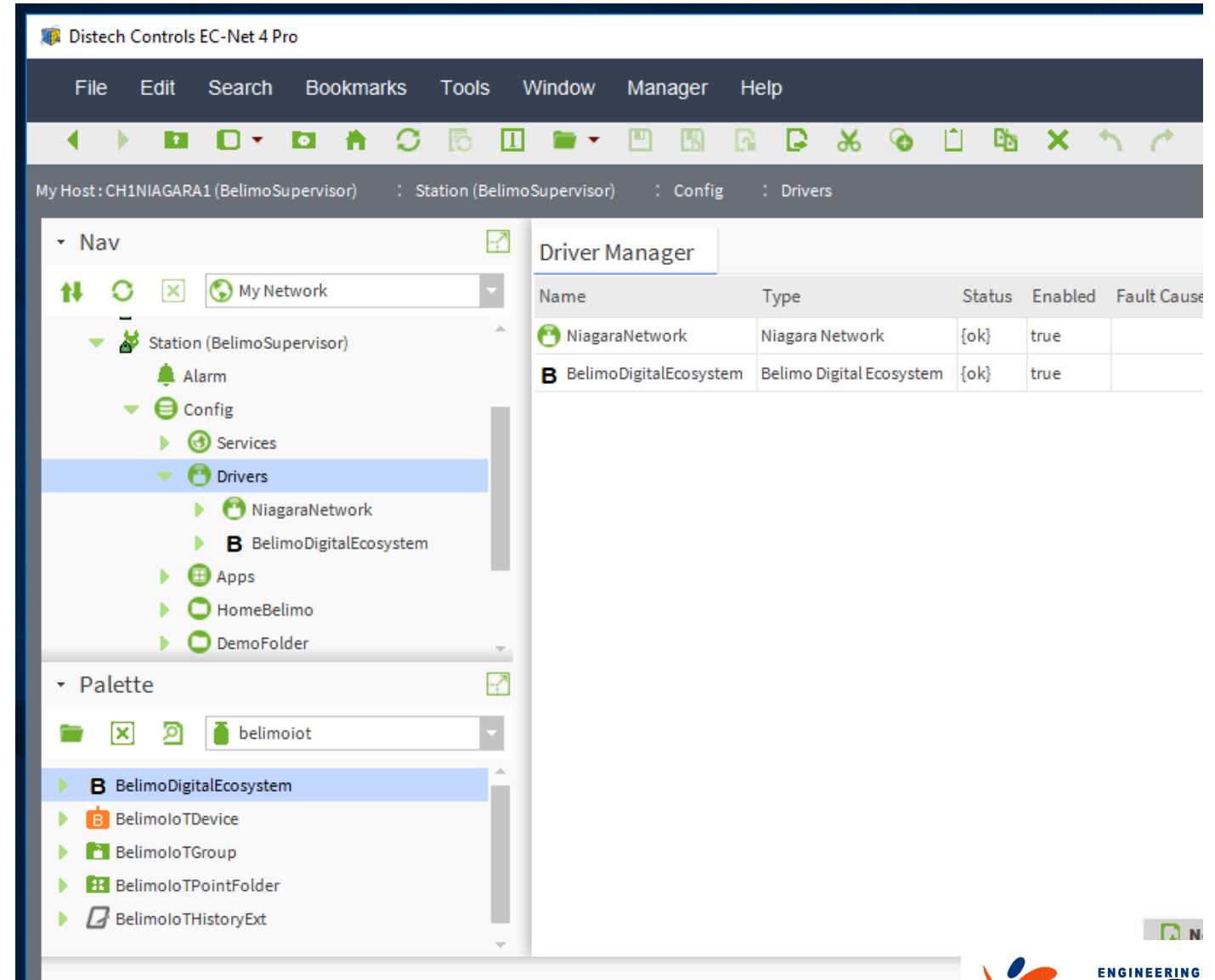
The Belimo palette

This is the palette of the Belimo Digital Ecosystem.



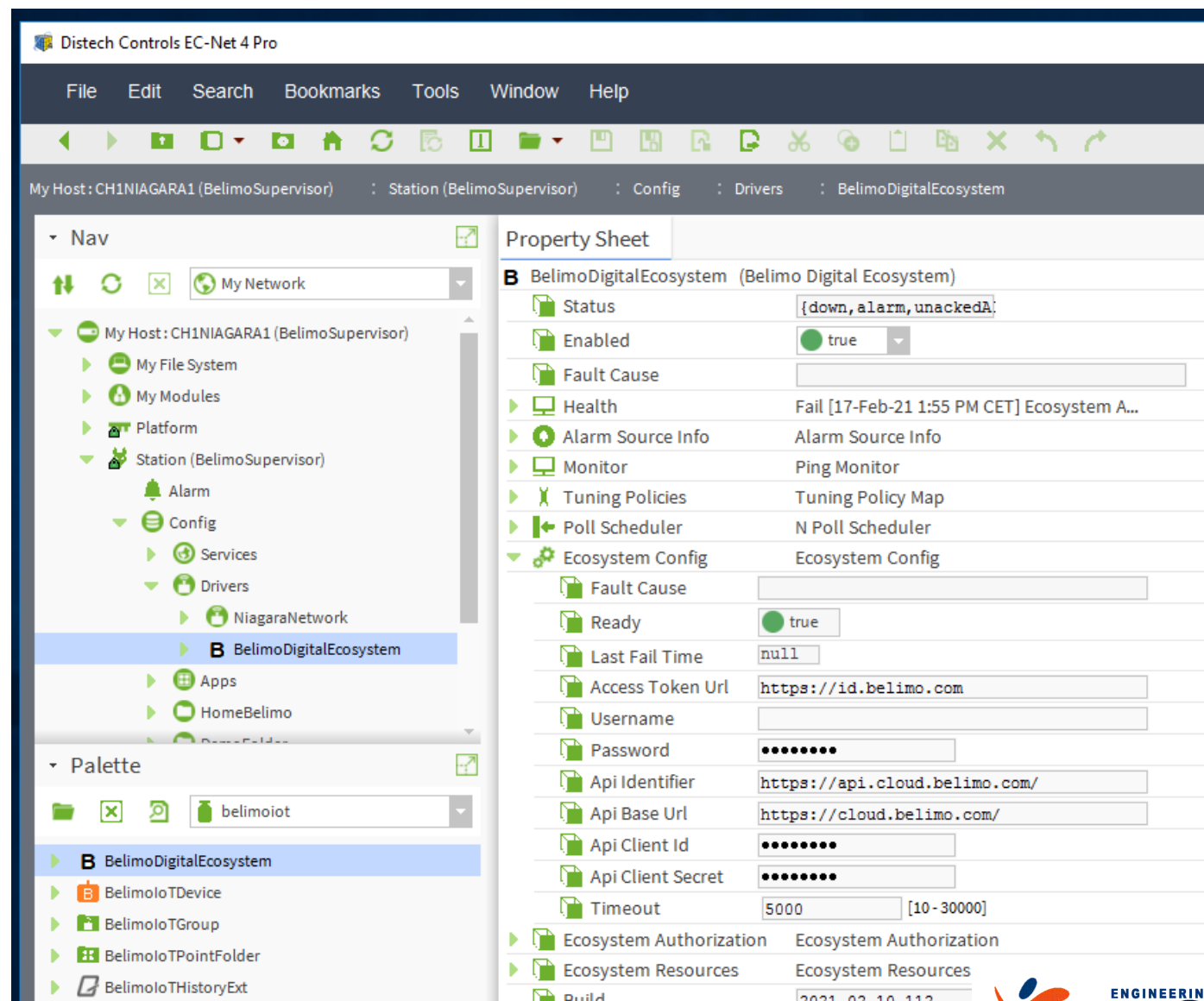
Add the driver to network

- Open the running station
- Navigate to Driver Manager
- Drop the 'BelimoDigitalEcosystem' object to Driver Manager



Configuration

- Select the network driver just added
- Open an AX Property Sheet View
- The status is not 'ok' as far as connection to Belimo cloud fails
- Expand the node 'Ecosystem Config'
- Input Username and password of your Belimo account.
- Input the Cloud API Client ID and Secret you received on the Belimo Developer Space (<https://www.belimo.com/iot/developers>)
- Restart the station



Connection established

Status 'ok' means connection to the Belimo Digital Ecosystem has been established successfully. Health indicator can produce more details about connection status.

...rs/BelimoDigitalEcosystem		Application Director		BelimoDigitalEcos	
Property Sheet					
BelimoDigitalEcosystem (Belimo Digital Ecosystem)					
Status	{ok}				
Enabled	<input checked="" type="radio"/> true				
Fault Cause					
Health	Ok [17-Feb-21 2:05 PM CET]				
Down	<input type="radio"/> false				
Alarm	<input type="radio"/> false				
Last Ok Time	17-Feb-2021 02:05 PM CET				
Last Fail Time	17-Feb-2021 02:03 PM CET				
Last Fail Cause	Ecosystem Authorization is missing				
Alarm Source Info	Alarm Source Info				








Device discovery

- Double click on the BelimoDigitalEcosystem network driver.
- The Belimoiot Device Manager view is showed
- Press the Discover button
- The list of devices is going to be updated according the devices associated to your Belimo account.

Supervisor) : Config : Drivers : BelimoDigitalEcosystem

Belimoiot Discovery

Discovered

Deviceid	Devicename	Ownerid	Ownername	Projectname
 a5690dde-be6f-46fd-983e-e9beb3aa7d00	PublicIoT Actuator V1.02	IA1Q808HY3	ClientAPI Usergroup	TEST
 108a36f6-6466-4988-b9b0-a3efd2005c42	EV54_EVNG_Feldtest_22011-30054-022-246	GZSGW1WQER	EnergyValve_Belimo	Project name
 ff265b95-ff7b-4d65-952b-f7087862df05	EV53_EVNG_Feldtest_22011-30053-022-246	GZSGW1WQER	EnergyValve_Belimo	Project name
 02f4e962-922e-499f-9e31-90210a6e17da	EV52_EVNG_Feldtest_22011-30052-022-246	GZSGW1WQER	EnergyValve_Belimo	Project name
 0c5aa95d-509e-4d66-b53c-5e2883b850d1	Heute LD5 670	GZSGW1WQER	EnergyValve_Belimo	Project name
 73ffd901-27b6-464a-a37b-2c717c3d36ab	EV4_EVNG_ERGON-TB_A3	GZSGW1WQER	EnergyValve_Belimo	Project name
 6f661025-ac12-46e0-9fe3-756fbd183227	EV4_EVNG_Feldtest_IER03	GZSGW1WQER	EnergyValve_Belimo	

Database

Name	Type	Exts	Status	Deviceid	Devicename	Ownerid	Ownername	Projectname	Addresscity	Addresscountry	Serialnumber	Pro
------	------	------	--------	----------	------------	---------	-----------	-------------	-------------	----------------	--------------	-----

New Folder New Edit Discover Cancel Add

Device add

- Select any device you would like to add then press button Add.

The 'Add' dialog box contains a table of existing devices and a form for adding a new one.

Name	Type	Status	Deviceid	Devicename
BelimoIoTDevice	Belimo Io T Device	{ok}	ff265b95-ff7b-4d65-952b-f7087862df05	EV53_EVNG_Feldtest_22011-30053-022-246
BelimoIoTDevice1	Belimo Io T Device	{ok}	02f4e962-922e-499f-9e31-90210a6e17da	EV52_EVNG_Feldtest_22011-30052-022-246

Below the table, the form fields are as follows:

- Name:** BelimoIoTDevice
- Type:** Belimo Io T Device (dropdown)
- Status:** {ok}
- Deviceid:** ff265b95-ff7b-4d65-952b-f7087862df05
- Devicename:** EV53_EVNG_Feldtest_22011-30053-022-246
- Ownerid:** GZSGW1WQER
- Ownername:** EnergyValve_Belimo
- Projectname:** Project name
- Addresscity:** City
- Addresscountry:** CH
- Serialnumber:** 22011-30053-022-246
- Profileid:** energyvalve4/1.2
- Profilename:** device.belimo.energy-valve-Version4/1.2
- Profileref:** /definitions/dataprofiles/energyvalve4/1








At the bottom of the dialog are 'OK' and 'Cancel' buttons. Below the dialog is a toolbar with buttons: 'New', 'Edit', 'Discover', 'Cancel', 'Add', 'Match', and 'TagIt'.

Device add





- Added devices are visible on lower list.

Belimoiot Discovery

Discovered

Deviceid	Devicename	O
 a5690dde-be6f-46fd-983e-e9beb3aa7d00	Public IoT Actuator V1.02	IA
 108a36f6-6466-4988-b9b0-a3efd2005c42	EV54_EVNG_Feldtest_22011-30054-022-246	G2
 ff265b95-ff7b-4d65-952b-f7087862df05	EV53_EVNG_Feldtest_22011-30053-022-246	G2
 02f4e962-922e-499f-9e31-90210a6e17da	EV52_EVNG_Feldtest_22011-30052-022-246	G2
 0c5aa95d-509e-4d66-b53c-5e2883b850d1	Heute LD5 670	G2
 73ffd901-27b6-464a-a37b-2c717c3d36ab	EV4_EVNG_ERGON-TB_A3	G2
 6f661025-ac12-46e0-9fe3-756fbd183227	EV4_EVNG_Feldtest_IER03	G2

Database

Name	Type	Exts	Status	Deviceid	Devicename
 BelimoloTDevice	Belimo lo T Device		{ok}	ff265b95-ff7b-4d65-952b-f7087862df05	EV53_
 BelimoloTDevice1	Belimo lo T Device		{ok}	02f4e962-922e-499f-9e31-90210a6e17da	EV52_

Device details

- Go to Nav Tree and double click one device
- A list of information is displayed.
- Status should be 'ok'
- Double click on the Points folder to go to Point Manager.

The screenshot displays the Belimo Digital Ecosystem interface. On the left, the 'Nav' (Navigation) tree shows a hierarchy starting from 'My Network' down to 'My Host: CH1NIAGARA1 (BelimoSupervisor)', then 'Platform', 'Station (BelimoSupervisor)', 'Alarm', 'Config', 'Services', 'Drivers', 'NiagaraNetwork', and finally 'BelimoDigitalEcosystem'. Under 'BelimoDigitalEcosystem', there are folders for 'Ecosystem Config', 'Ecosystem Authorization', 'Ecosystem Resources', 'Monitor', 'Tuning Policies', and two device entries: 'BelimoloTDevice' (highlighted in blue) and 'BelimoloTDevice1'. On the right, the 'Property Sheet' for 'BelimoloTDevice (Belimo Io T Device)' is shown. It contains various properties and their values:








Property Sheet	
BelimoloTDevice (Belimo Io T Device)	
Status	{ok}
Enabled	true
Fault Cause	
Health	Fail [null]
Alarm Source Info	Alarm Source Info
Poll Frequency	Normal
Points	Belimo Io T Point D
Last Cloud Read	17-Feb-2021 02:3
Deviceid	ff265b95-ff7b-4c
Devicename	EV53_EVNG_Feldte
Ownerid	GZSGW1WQER
Ownername	EnergyValve_Beli
Projectname	Project name
Addresscity	City
Addresscountry	CH
Serialnumber	22011-30053-022-
Profileid	energyvalve4/1.2
Profilename	device.belimo.er
Profileref	/definitions/dat

Data Points selection

- When the Point Manager view of a selected device is showed you can press the Discovery button.
- A list of available data points is filled.
- Select the points you would like using and press Add.

BelimoIot Discovery

Discovered

Pointid	Point Caption	Point Name	Current Value	Description
 evcloud.00009	Valve Type (0=2-way EV3, 1=3-way EV3)	ValveType	V0	Valve Type
 evcloud.200	Cooling Energy	Cooling_E_J	2.9623251385726845E10	Cooling Ene
 evcloud.180	Setpoint DeltaT	SpDeltaT_applied_K	10.0	Applied Setp
 evcloud.170	Delta T Limitation Type	DeltaT_Limitation_Write	V2	Delta T Limit
 evcloud.190	Delta Temperature	DeltaT_K	24.329999999999984	Delta Tempe
 evcloudplus.384	Delta T Manager Active	Active_dT_Manager_total_h	1644.6273239298453	while_DDC_!
 evcloud.80	Relative Position	RelPos_Fb	52.49	Relative Pos

Database

Name	Type	Out	Pointid	Can Write
------	------	-----	---------	-----------

New Folder New Edit Discover Cancel

Data Points added

- Added points are visible on Nav Tree too.

The screenshot displays the Belimo Digital Ecosystem interface. On the left is the 'Nav Tree' (Navigation Tree) showing a hierarchical structure of the system. The 'BelimoTDevice' folder is expanded, revealing 'Alarm Source Info', 'Points', and 'BelimoTDevice1'. Under 'Points', two data points are listed: 'SpDeltaT_applied_K' and 'DeltaT_K'. On the right, the 'BelimoTDiscovery' window is open, showing two tabs: 'Discovered' and 'Database'.

Discovered Table:

Pointid	Point Caption	Point Name
evcloud.00009	Valve Type (0=2-way EV3, 1=3-way EV3)	ValveType
evcloud.200	Cooling Energy	Cooling_E_J
evcloud.180	Setpoint DeltaT	SpDeltaT_appli
evcloud.170	Delta T Limitation Type	DeltaT_Limitati
evcloud.190	Delta Temperature	DeltaT_K
evcloudplus.384	Delta T Manager Active	Active_dT_Man
evcloud.80	Relative Position	RelPos_Fb

Database Table:

Name	Type	Out	Pointid	Can Write
SpDeltaT_applied_K	B NumericPoint	10.0 {ok}	evcloud.180	false
DeltaT_K	B NumericPoint	25.1 {ok}	evcloud.190	false

Data Point details

- Double click on added data point and a details view is showed.
- The current value is available too.

The screenshot displays the Belimo Digital Ecosystem software interface. The top navigation bar shows the current host: CH1NIAGARA1 (BelimoSupervisor), and the selected view: Station (BelimoSupervisor). The left sidebar contains a navigation tree with the following structure:

- My Host: CH1NIAGARA1 (BelimoSupervisor)
 - My File System
 - My Modules
 - Platform
 - Station (BelimoSupervisor)
 - Alarm
 - Config
 - Services
 - Drivers
 - NiagaraNetwork
 - BelimoDigitalEcosystem
 - Ecosystem Config
 - Ecosystem Authorization
 - Ecosystem Resources
 - Monitor
 - Tuning Policies
 - BelimoIoTDevice
 - Alarm Source Info
 - Points
 - SpDeltaT_applied_K (B Numeric Point)
 - DeltaT_K

The right sidebar displays the Property Sheet for the selected data point, SpDeltaT_applied_K (B Numeric Point). The properties are as follows:

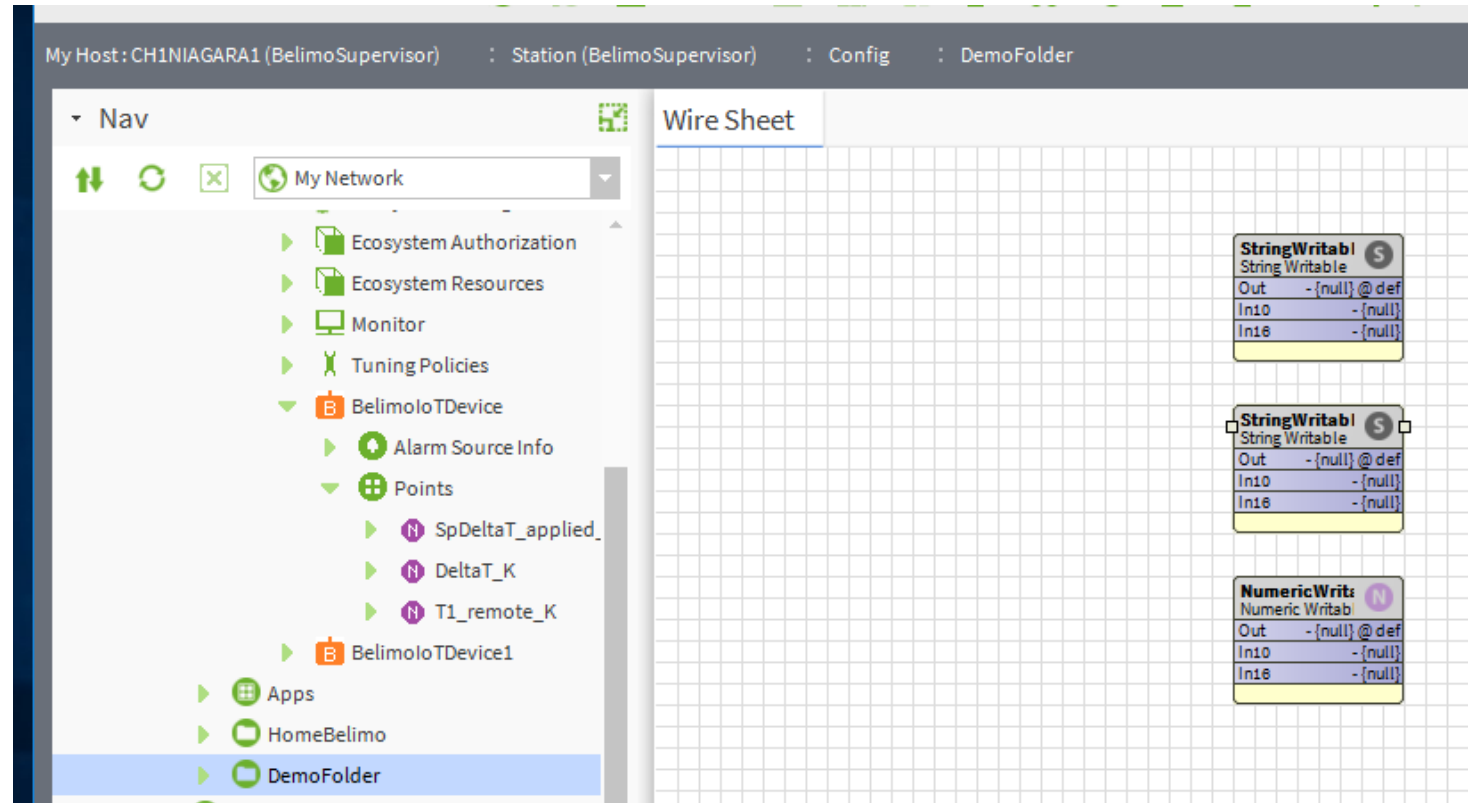
Property	Value
Facets	units=null,precision=1,min=-inf,max=+inf
Proxy Ext	Belimo Io T Proxy Ext
Out	10.0 {ok}
Description	Applied Setpoint DeltaT in K
Point Caption	Setpoint DeltaT
Pointid	evcloud.180

Binding

Next steps show how to bind
point value to local variables

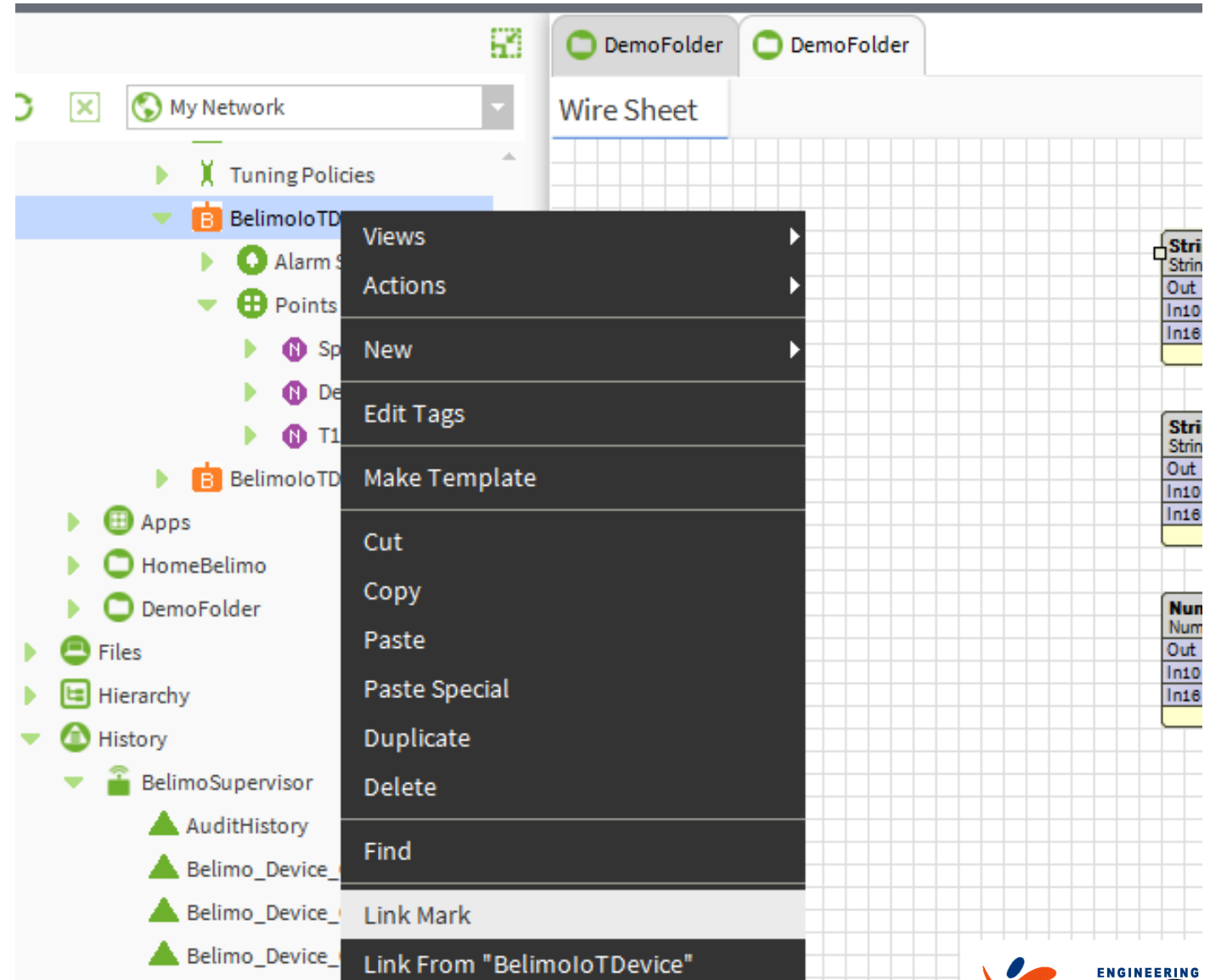
Demo folder sheet

- Create a folder, here for sample purposes is named 'DemoFolder'.
- Open the Sheet view
- Add on the sheet two string writable object
- Add on the sheet on numeric writable object



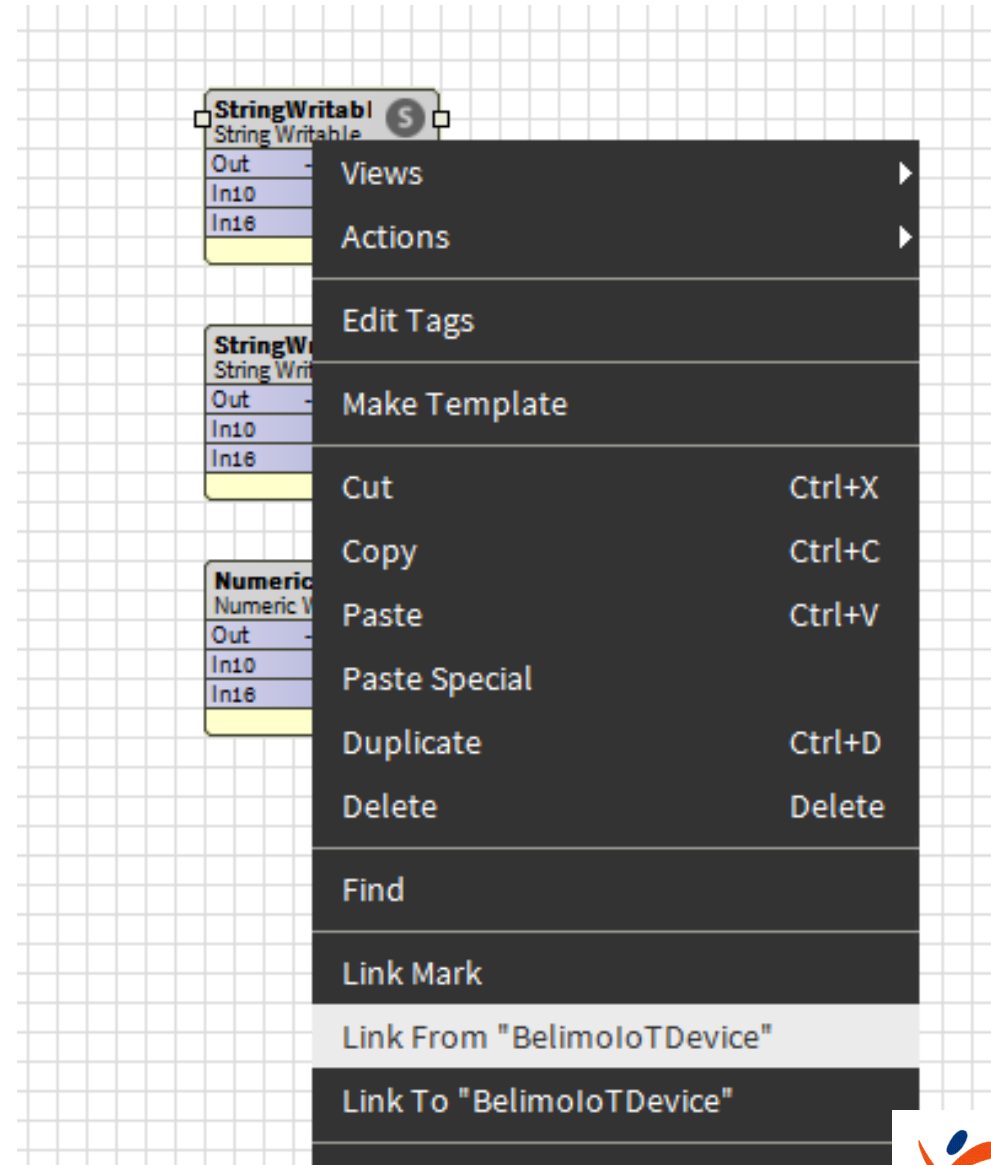
Bind from device

- Select the device and perform Link mark



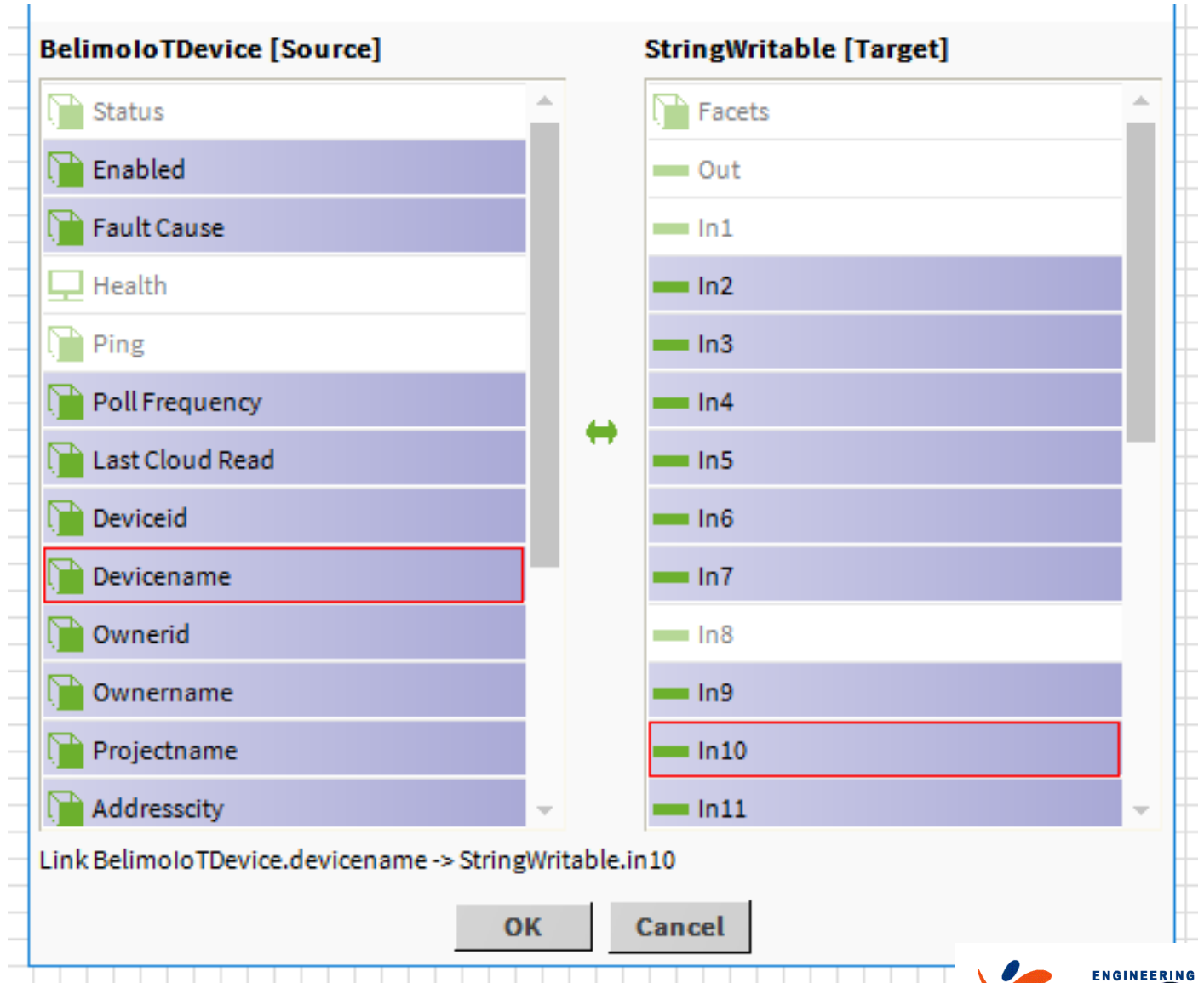
Bind from device

- Select first string writable object
- Perform a Link from “BelimolotDevice”



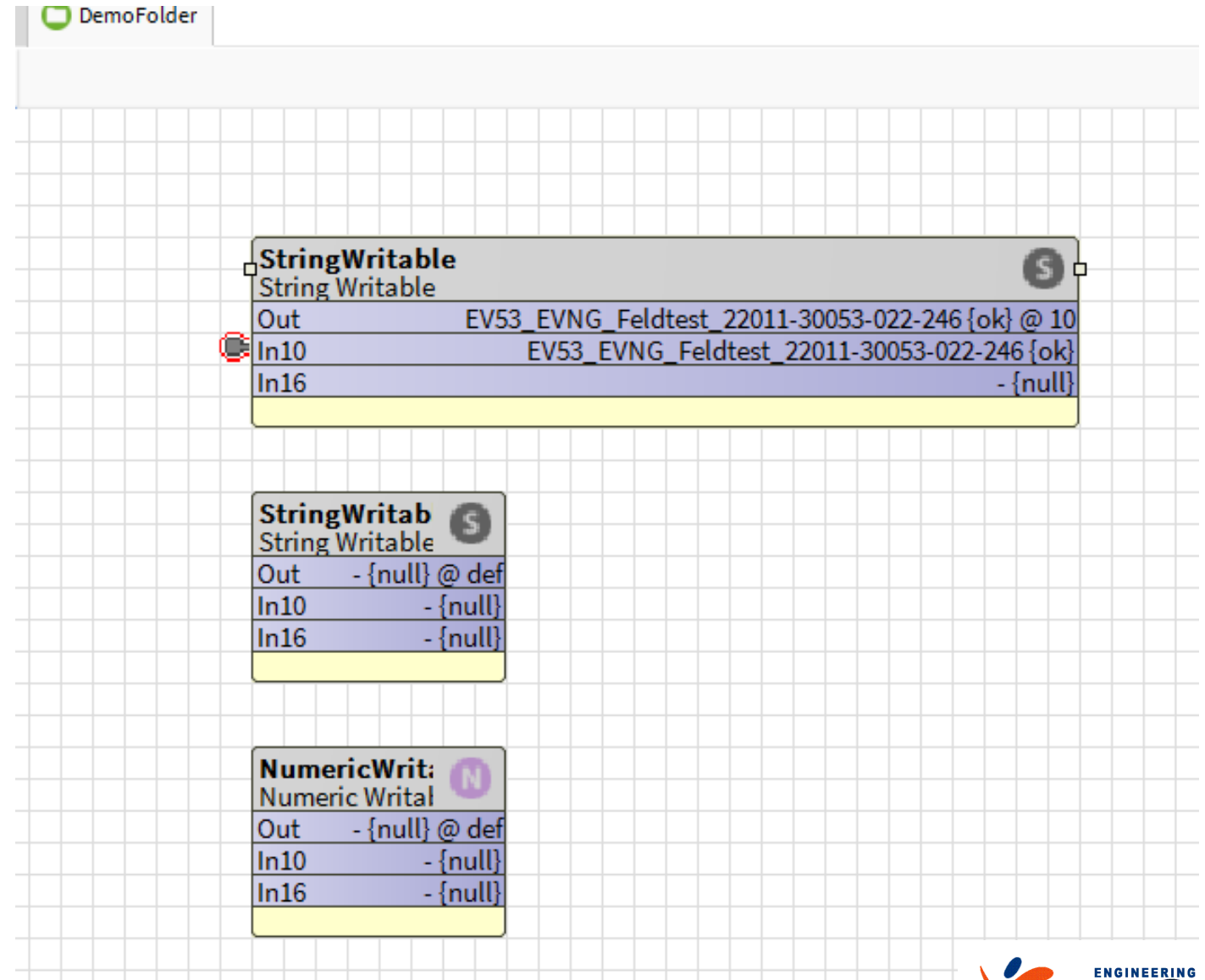
Bind from device

- Select source and target of binding as in the example



Devicename bind

- Now device name is available as variable



Data Point value

- Perform the same operation marking for link a point value out

The screenshot displays the Belimo Digital Ecosystem interface. On the left, a tree view shows the hierarchy of devices and points. The 'T1_remote_K' point is selected under the 'BelimoloTDevice' category. On the right, a 'Link' dialog box is open, showing a mapping between 'T1_remote_K [Source]' and 'NumericWritable [Target]'. The 'Out' facet of the source is linked to the 'In10' facet of the target. The dialog box also shows a list of facets for both source and target, and a status bar at the bottom indicates the link operation: 'Link T1_remote_K.out -> NumericWritable.in10'.

Belimo Digital Ecosystem by INLON

Data Point description

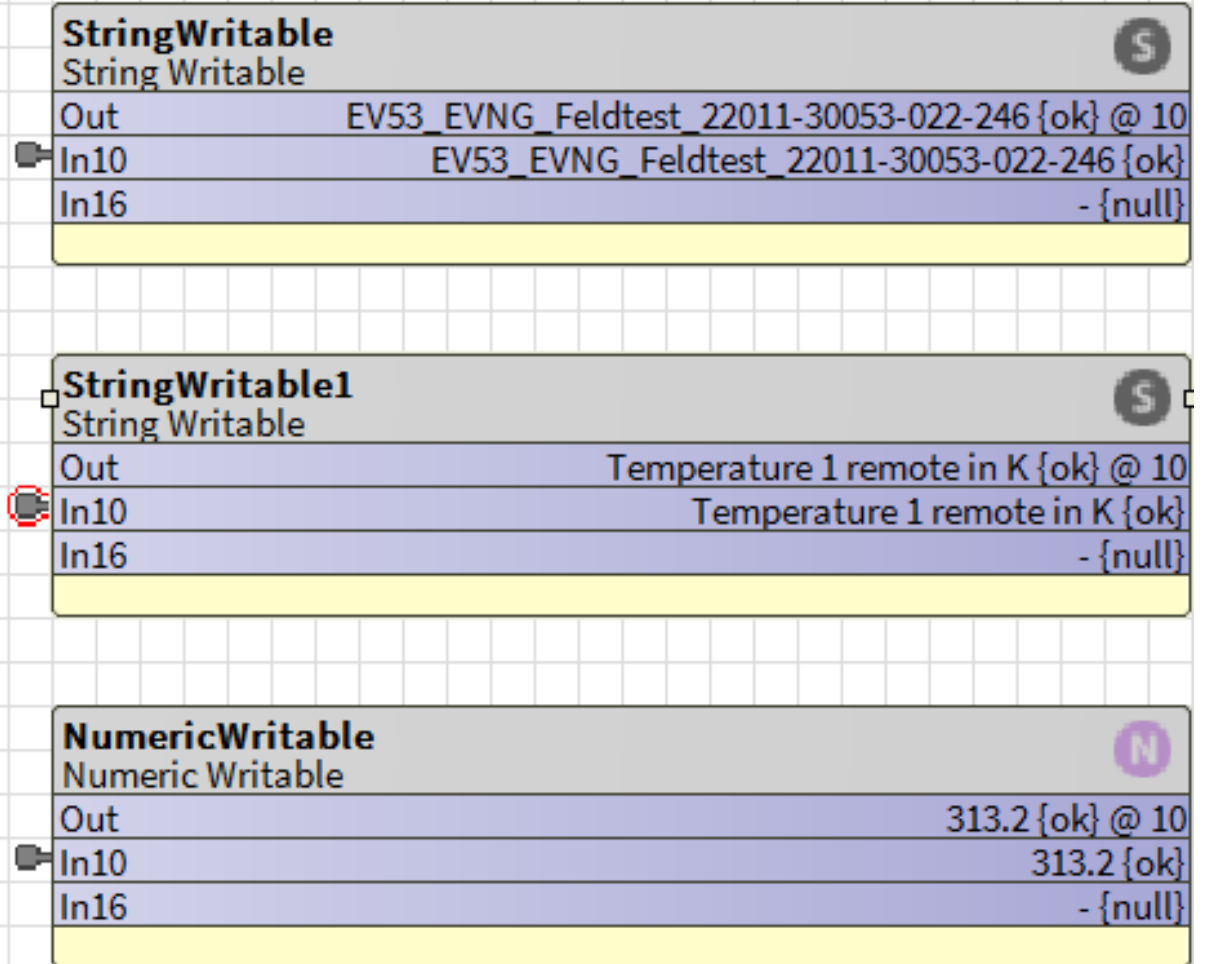
- Perform the same operation marking for link a point description

The screenshot displays the Belimo Digital Ecosystem software interface. On the left, a tree view shows the system hierarchy, with 'T1_remote_K' selected under the 'Points' category. The main area shows a 'Link' dialog box. The dialog has two panes: 'T1_remote_K [Source]' and 'StringWritable1 [Target]'. In the source pane, 'Description' is highlighted. In the target pane, 'In10' is highlighted. A double-headed arrow connects the two panes. At the bottom of the dialog, the text 'Link T1_remote_K.description -> StringWritable1.in10' is displayed. The background shows a grid pattern.

Belimo Digital Ecosystem by INLON

Variable bind

- Values and point information can be used to design your own pages



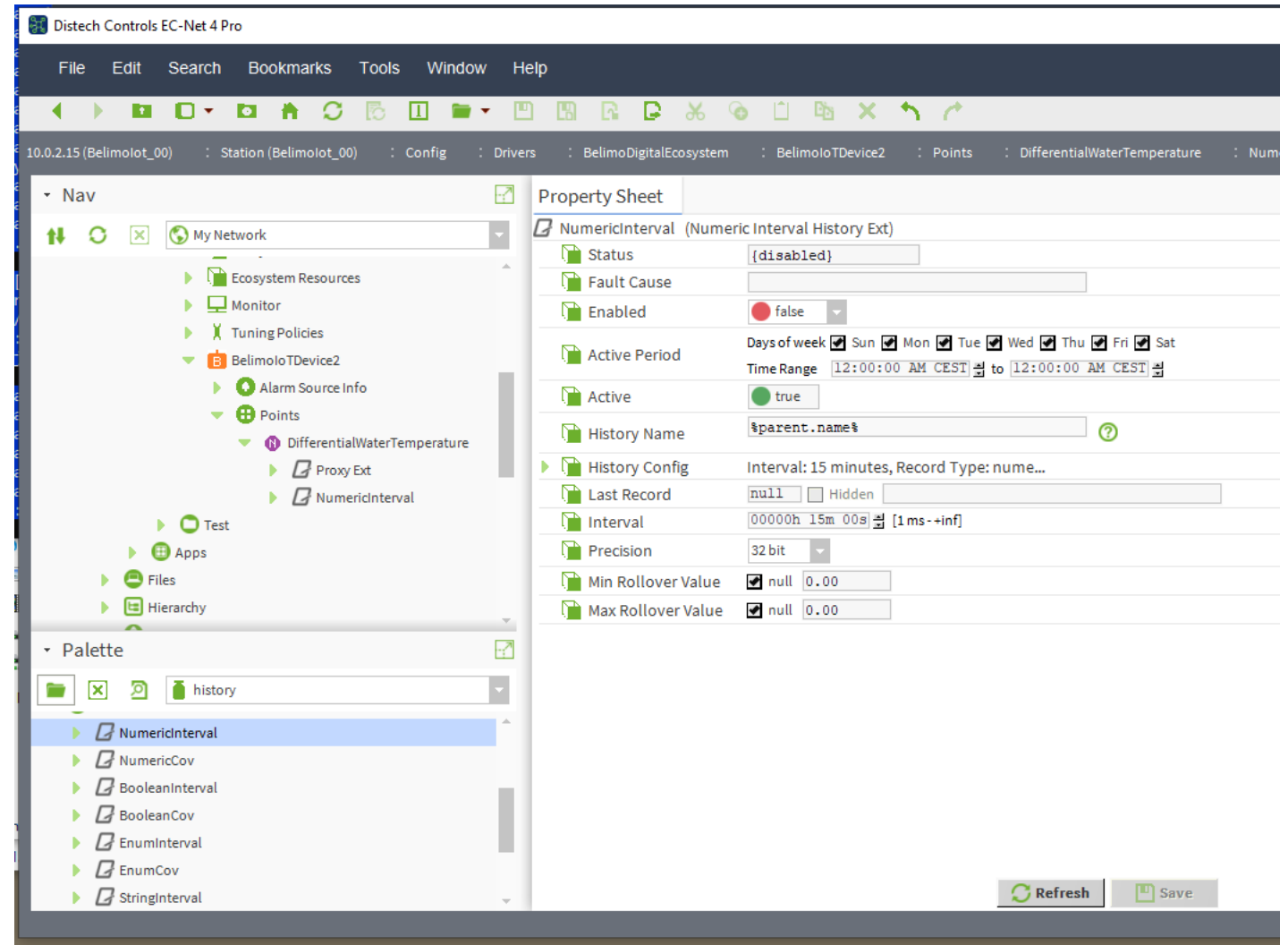
History

Next steps show how to get one point history data from cloud

History extensions

Add a history extension to any point to log historical data, that is, to collect a history. Standard components are found in the Extensions folder of the history palette. The Belimo Ecosystem supply a specific history extension to get history data from cloud.

Standard extensions collect data only if connection with devices is present.

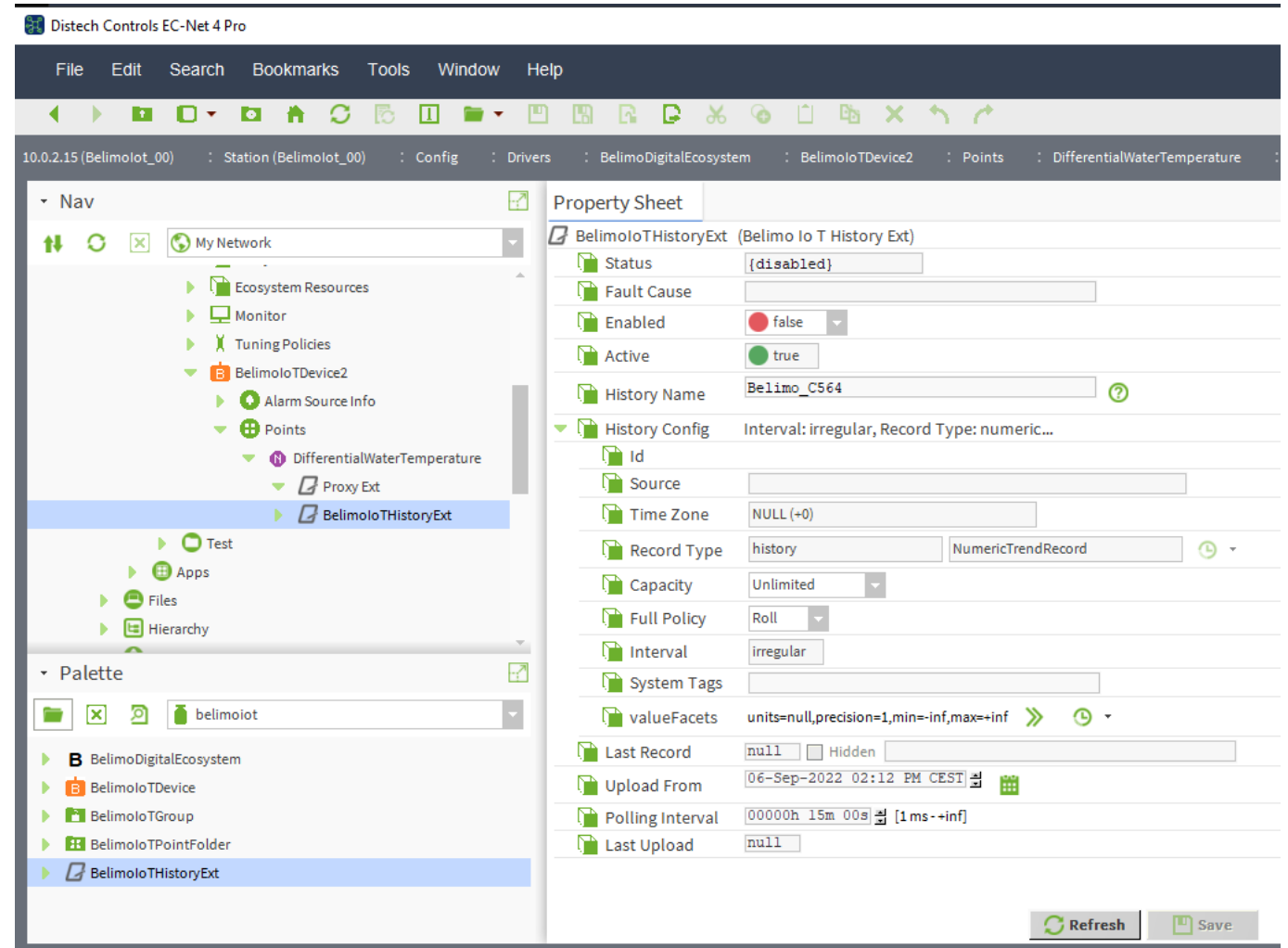


Add Belimo history extension

Belimo history extension allows to read data stored in the cloud. If the connection with cloud has missed, the extension get older data while the connection was missing.

- Select device point and show AX Property Sheet.
- From the palette panel drag and drop the BelimoTHistoryExt into Property Sheet.
- Expand the node just added

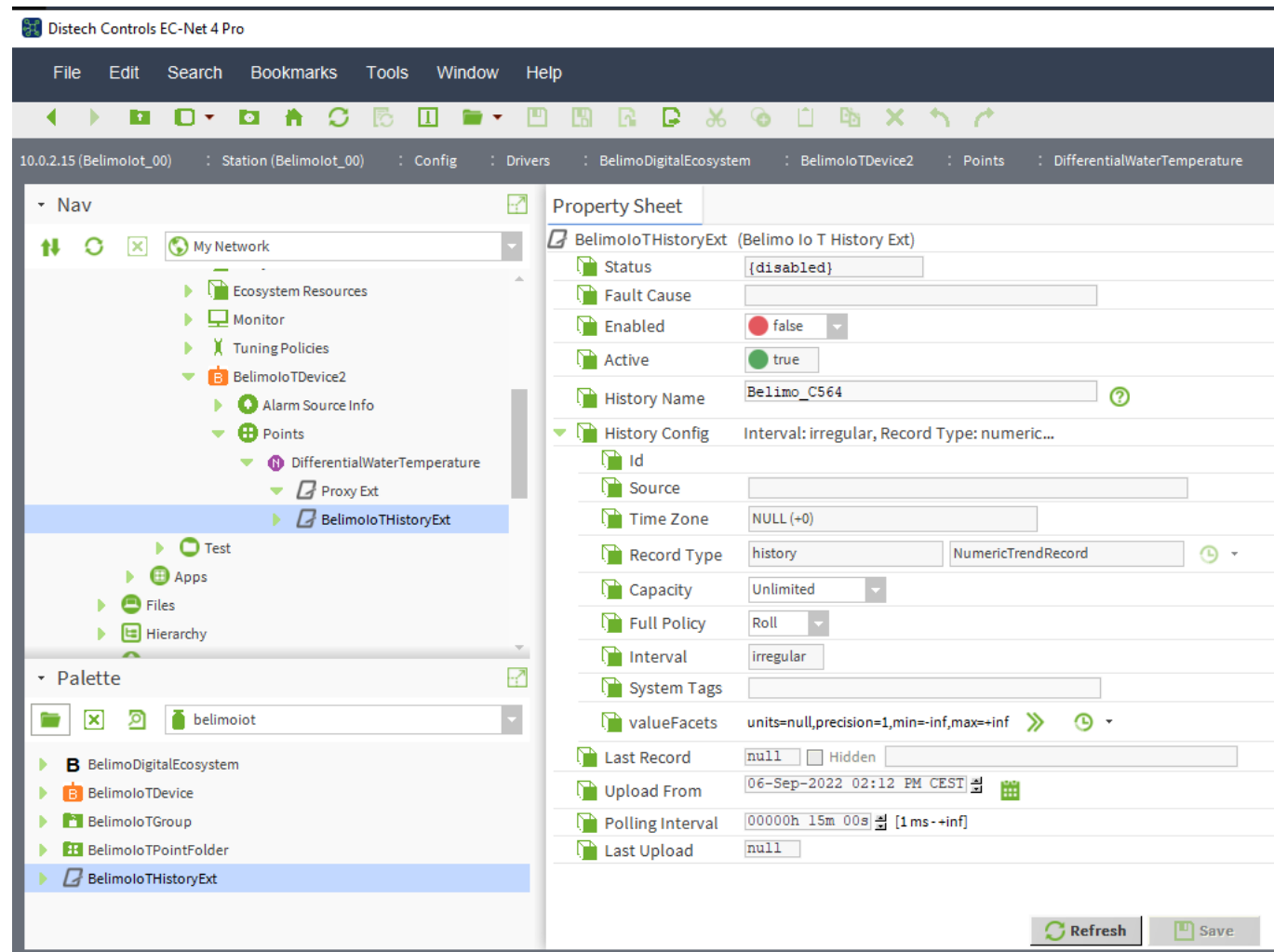
Belimo extension can coexist with standard extension.



Belimo history extension settings

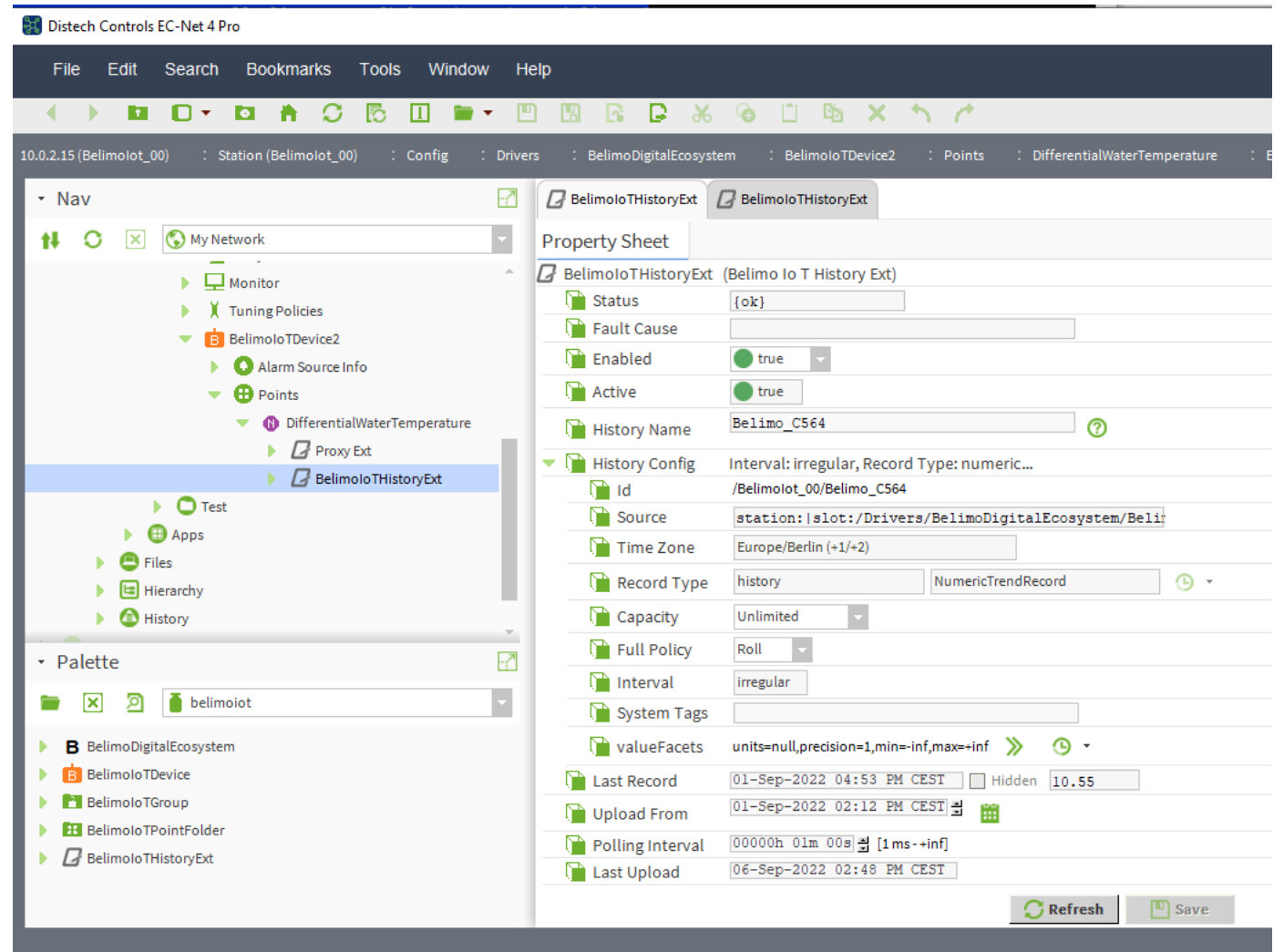
There are some important settings to apply on the component.

- History Name: the component default is a univocal short identifier. The default value can be substituted with “%parent.name%” like standard extensions.
- Upload From: the default value is the date and time when component has been added. A specific date can be selected to download older data at first activation.
- Polling interval: this is the interval time the component wait before to ask for new data to cloud service.



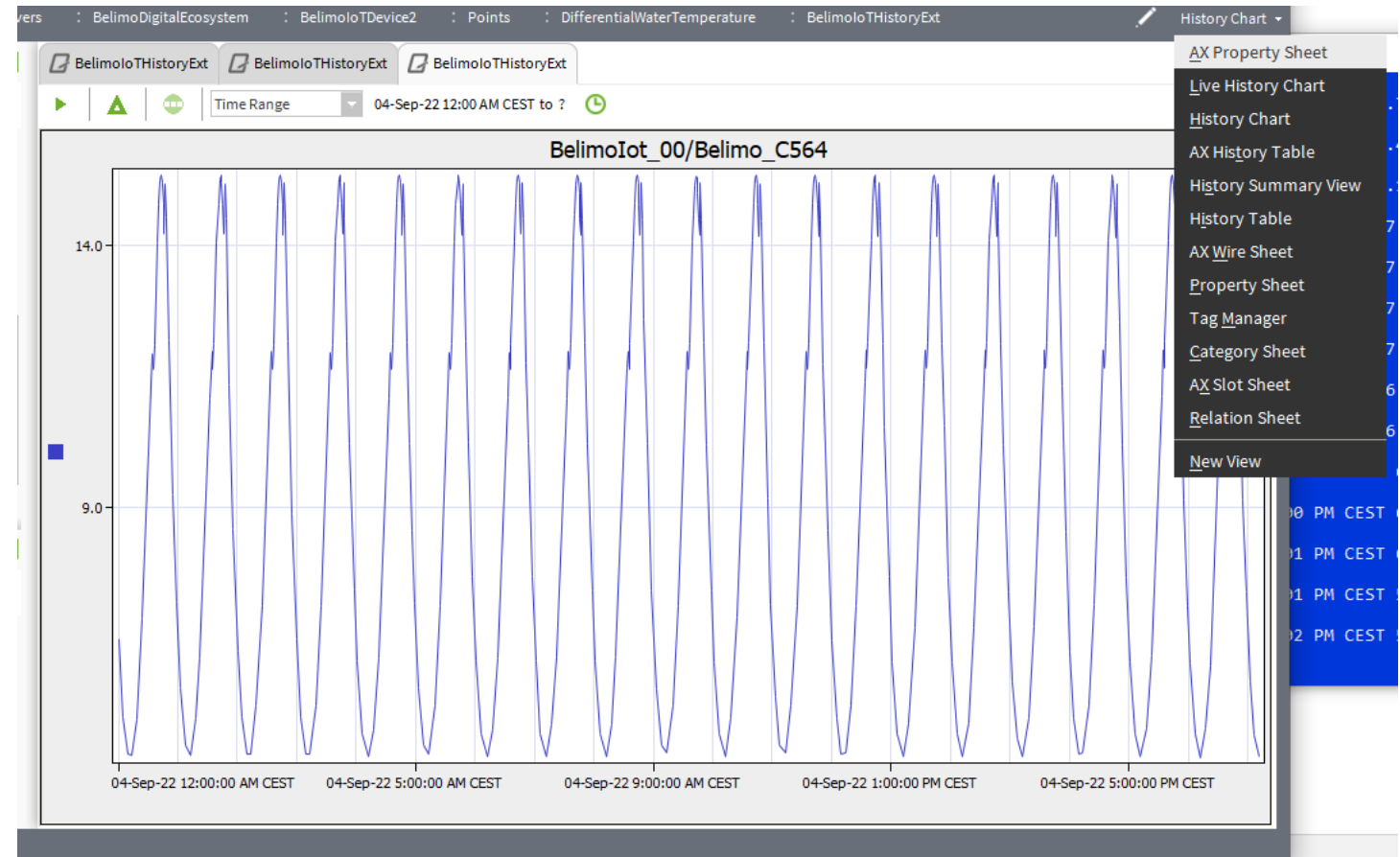
History activation

- Enable the history.
- After this operation an history is created.
- Data downloading can require long time before to complete.



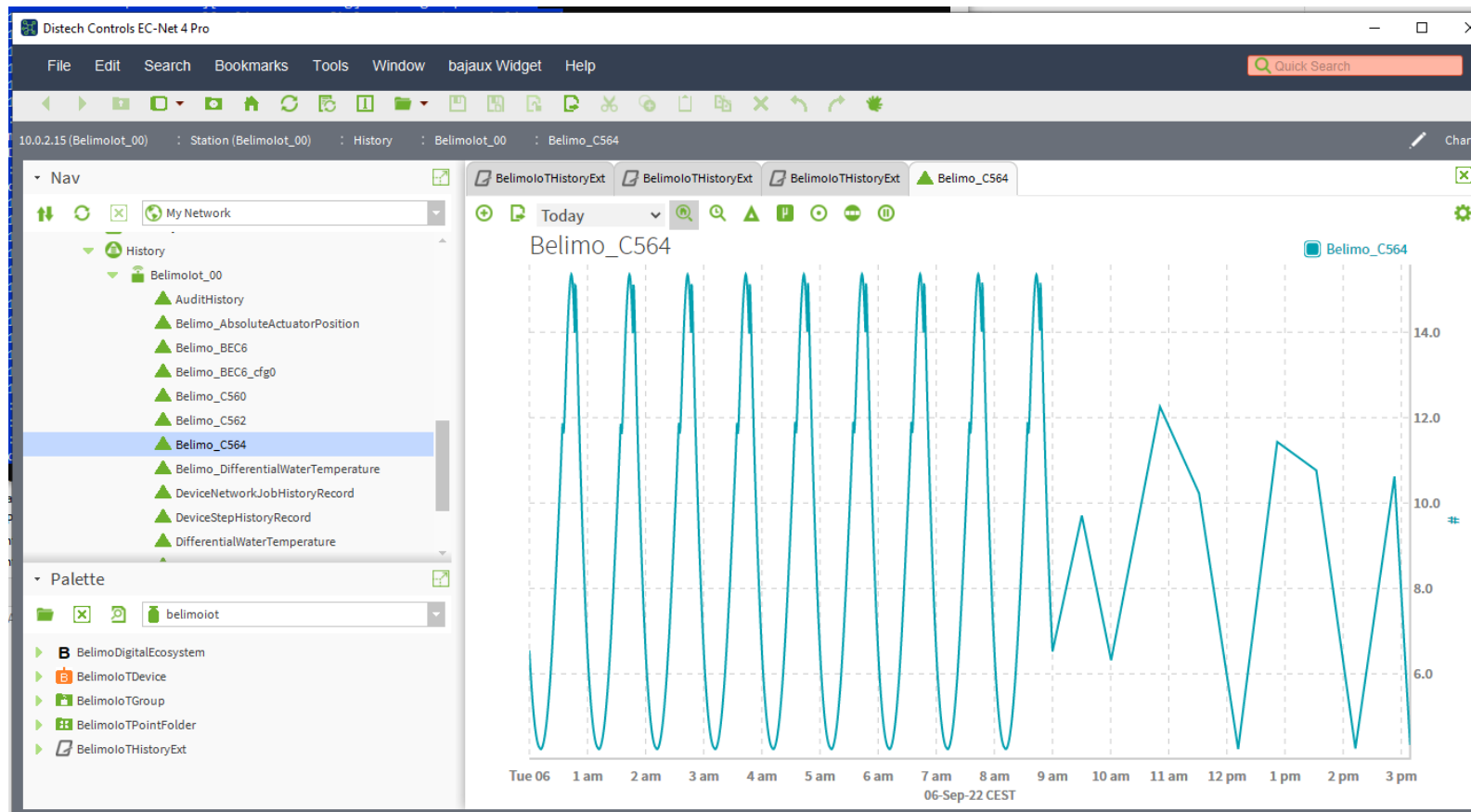
Access to history chart (1)

- Click on right panel History chart



Access to history (2)

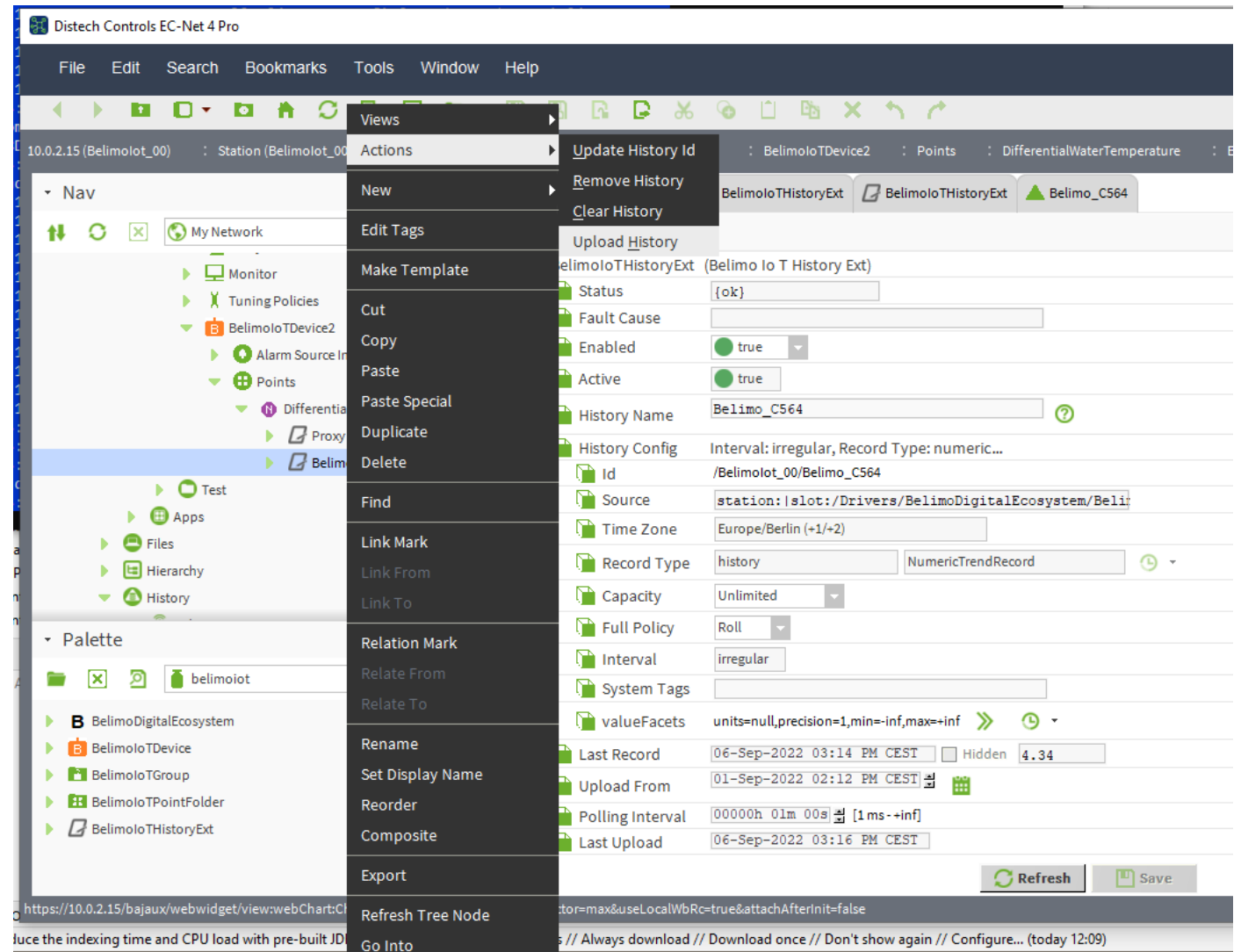
- Navigate to History main folder
- Search the item has the same id of your history point extension
- Double click on the item and a chart view is showed



Belimo History actions

The Belimo history extension has some specific action available.

- Remove History: delete the history database.
- Clear History: remove all data in the history repository.
- Upload History: get data from cloud immediately.



The connector has been developed by:
Inlon Engineering S.r.l.
Via Zara, 2
19123 La Spezia (ITALY)

Web site:
www.inlon.it

Contact:
info@inlon.it

Technical support:
mraymondi@inlon.it

You can download from:
github.com/inlon-engineering/niagara-belimo-iot

Thanks to:
BELIMO Automation AG

