JMobile Training Day 2

JM 4.0

Agenda

Manage Target

Remote Control

Interfaces

Corvina Cloud

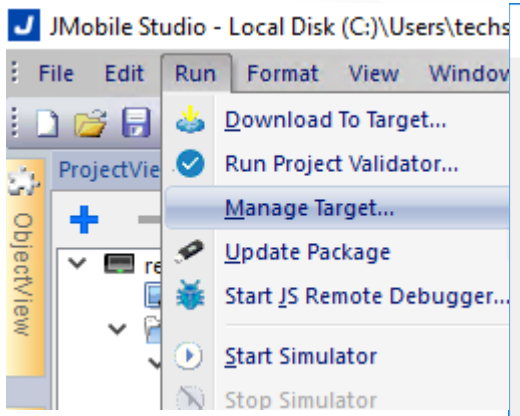
JavaScript

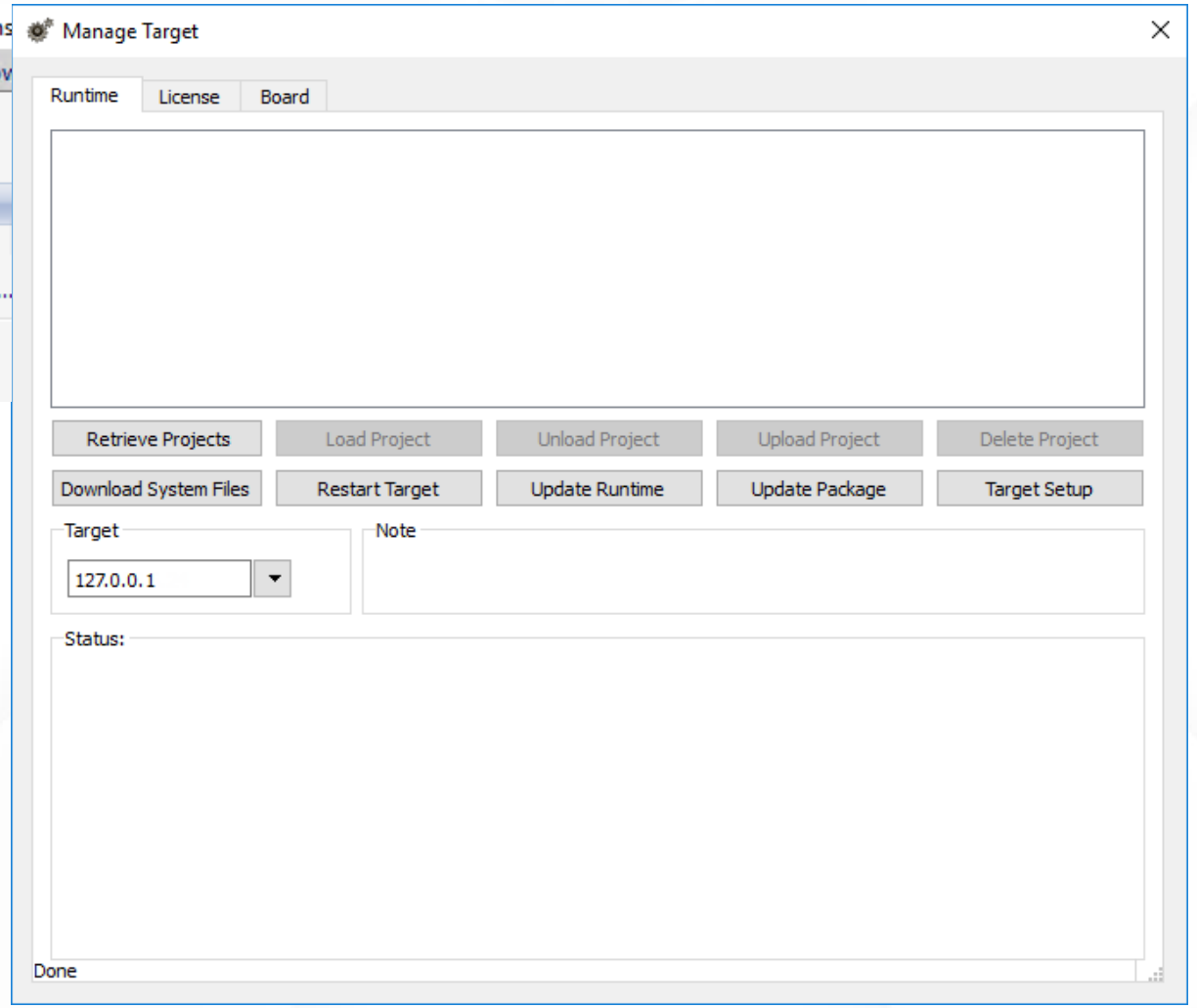
Custom Widgets and User Gallery

CODESYS internal PLC

Manage Target

Manage Target



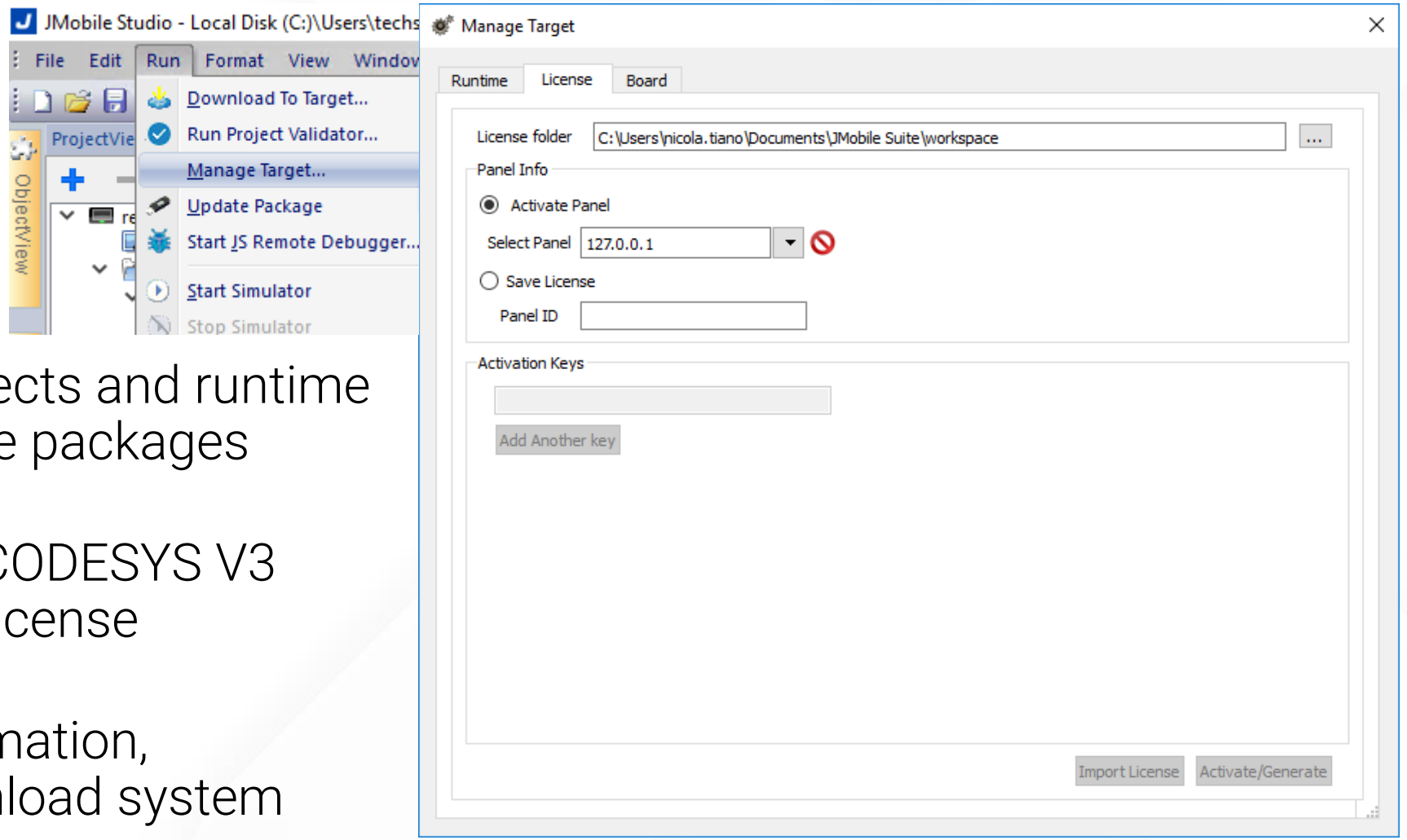


Runtime: manage projects and runtime create update packages

License: activation of CODESYS V3 internal PLC license

Board: get board information, upload/download system components

Manage Target

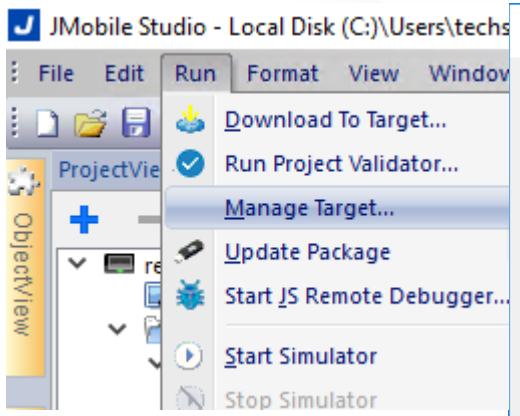


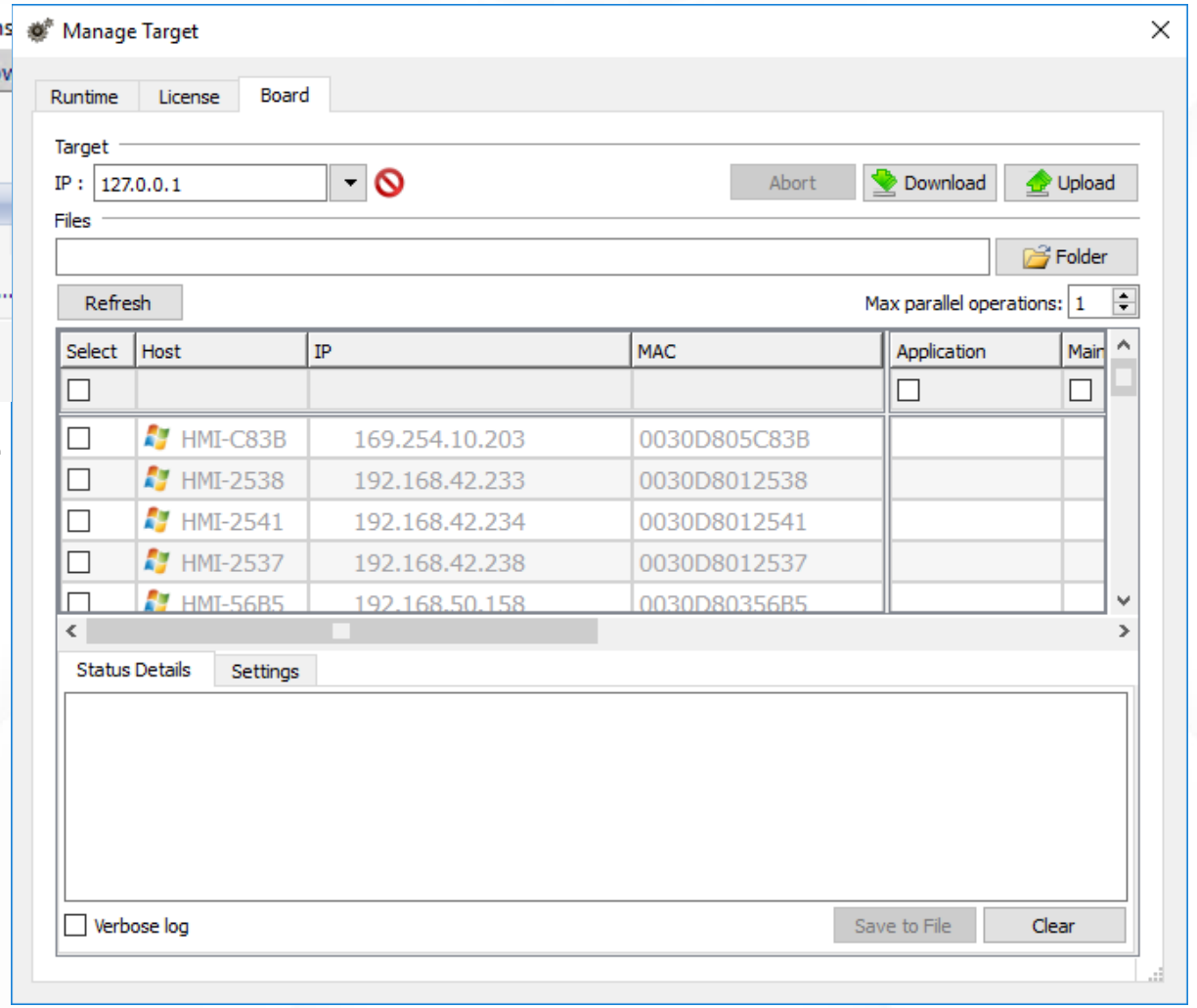
Runtime: manage projects and runtime create update packages

License: activation of CODESYS V3 internal PLC license

Board: get board information, upload/download system components

Manage Target





Runtime: manage projects and runtime
create update packages

License: activation of CODESYS V3
internal PLC license

Board: get board information,
upload/download system
components

Manage Target - Runtime

List of stored projects

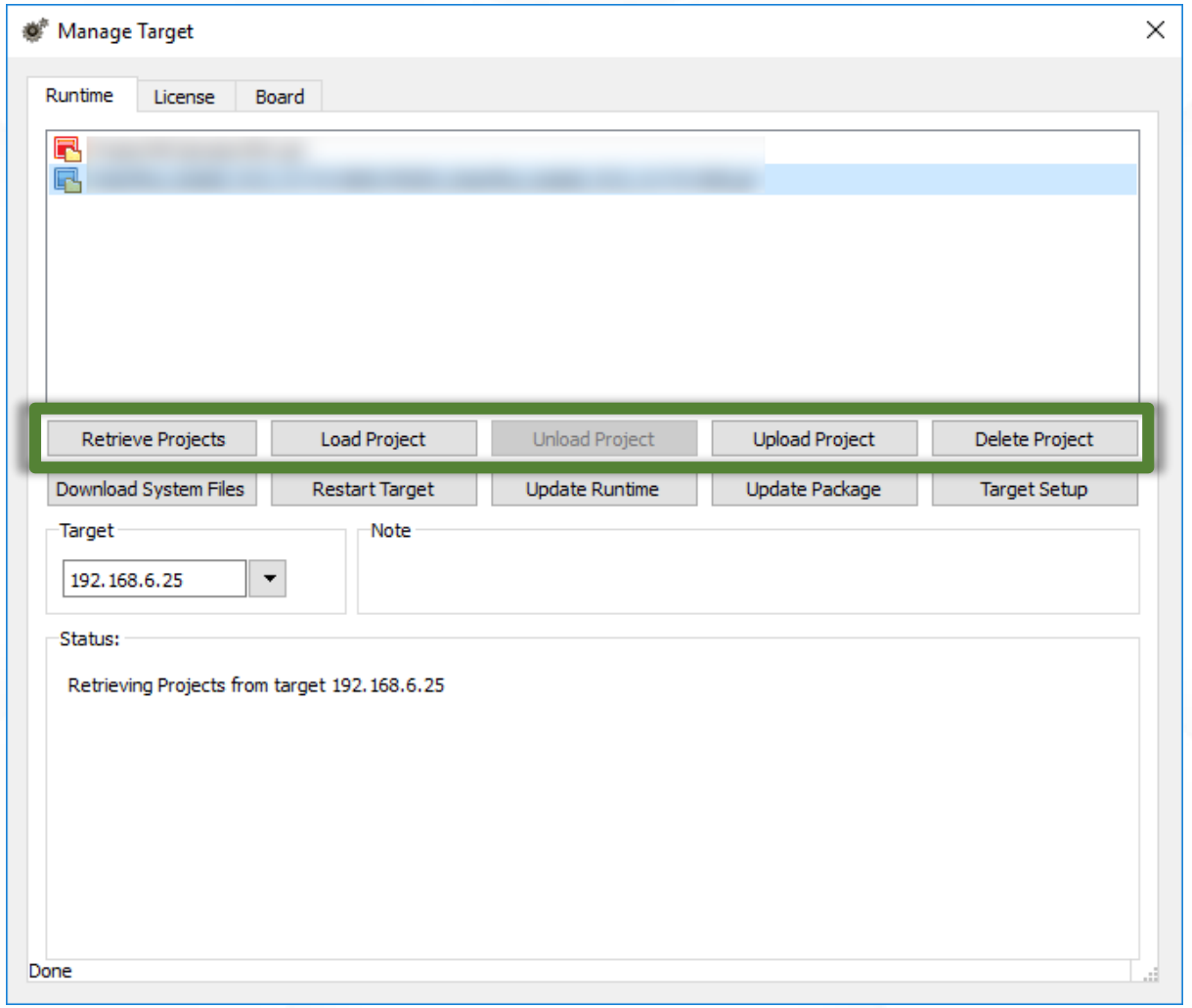
Load and unload projects

Red icon > project is unloaded

Blue icon > project is loaded

Delete projects to free space

Upload projects *



* Copy project from HMI to PC.
HMI Can be password-protected to prevent “unwanted” uploads

Manage Target - Runtime

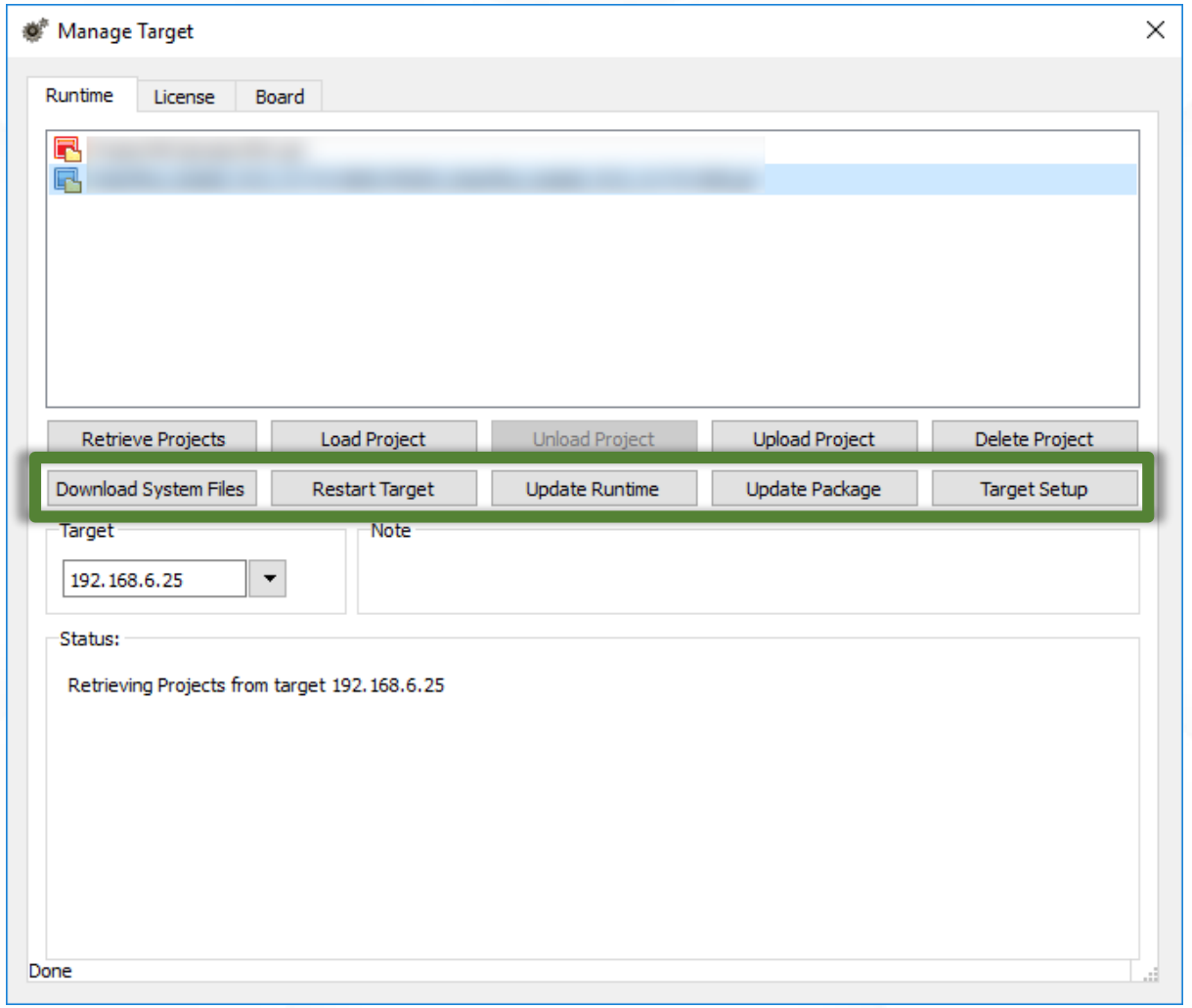
Update Runtime

Restart Target

Create Update Packages

Change target listening ports *

Download System Files to apply



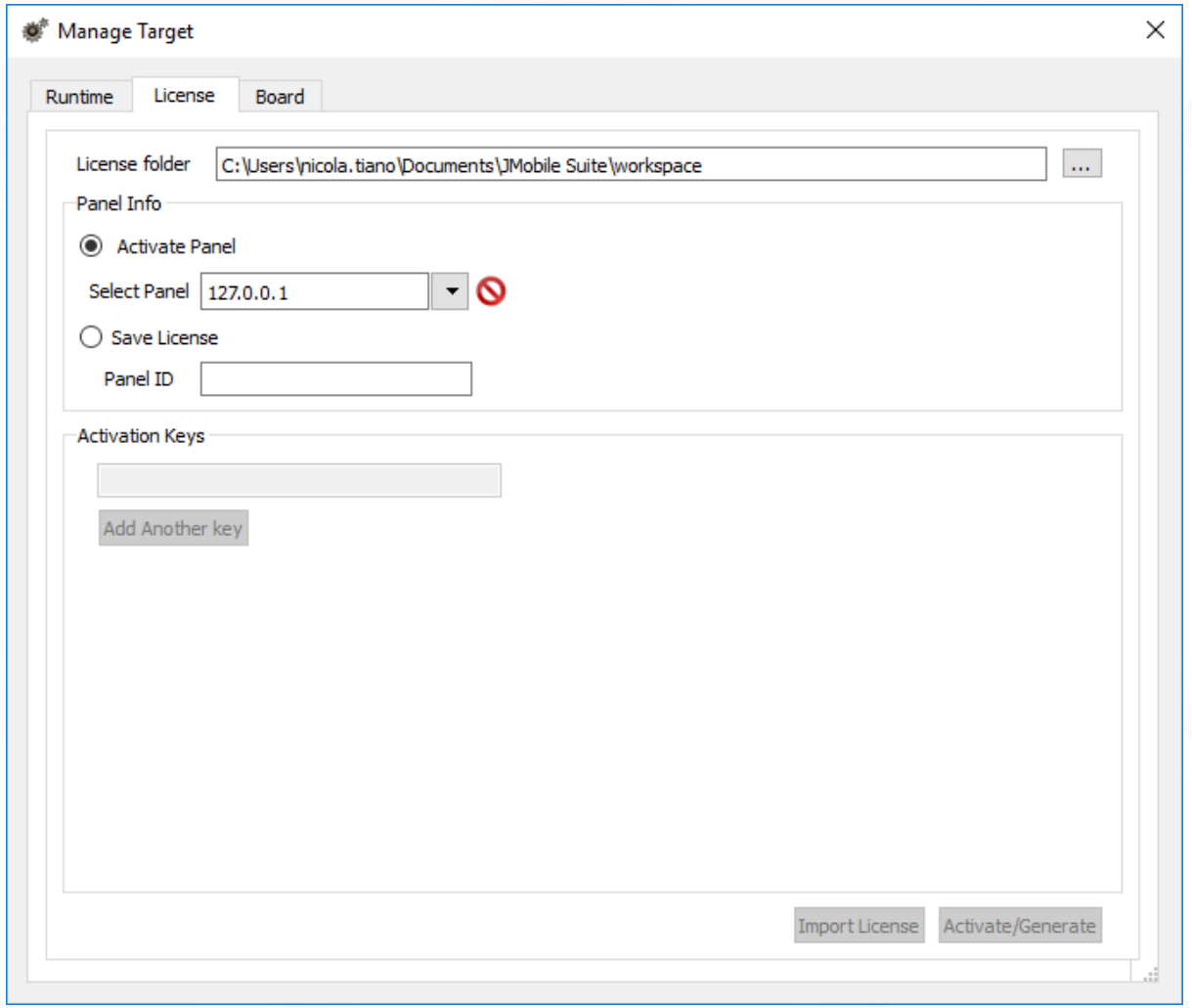
* Useful when target is Runtime PC and default HTTP port (80) is already used by other programs

Manage Target - License

Insert HMI IP Address
or search from list

Insert Activation Key *

Click on Activate/Generate



* Activation Key can be:

Final License: ask to sales

Demo License: insert “CODESYS_DEMO” to have a 2 hours running PLC. To restart demo license, HMI requires reboot

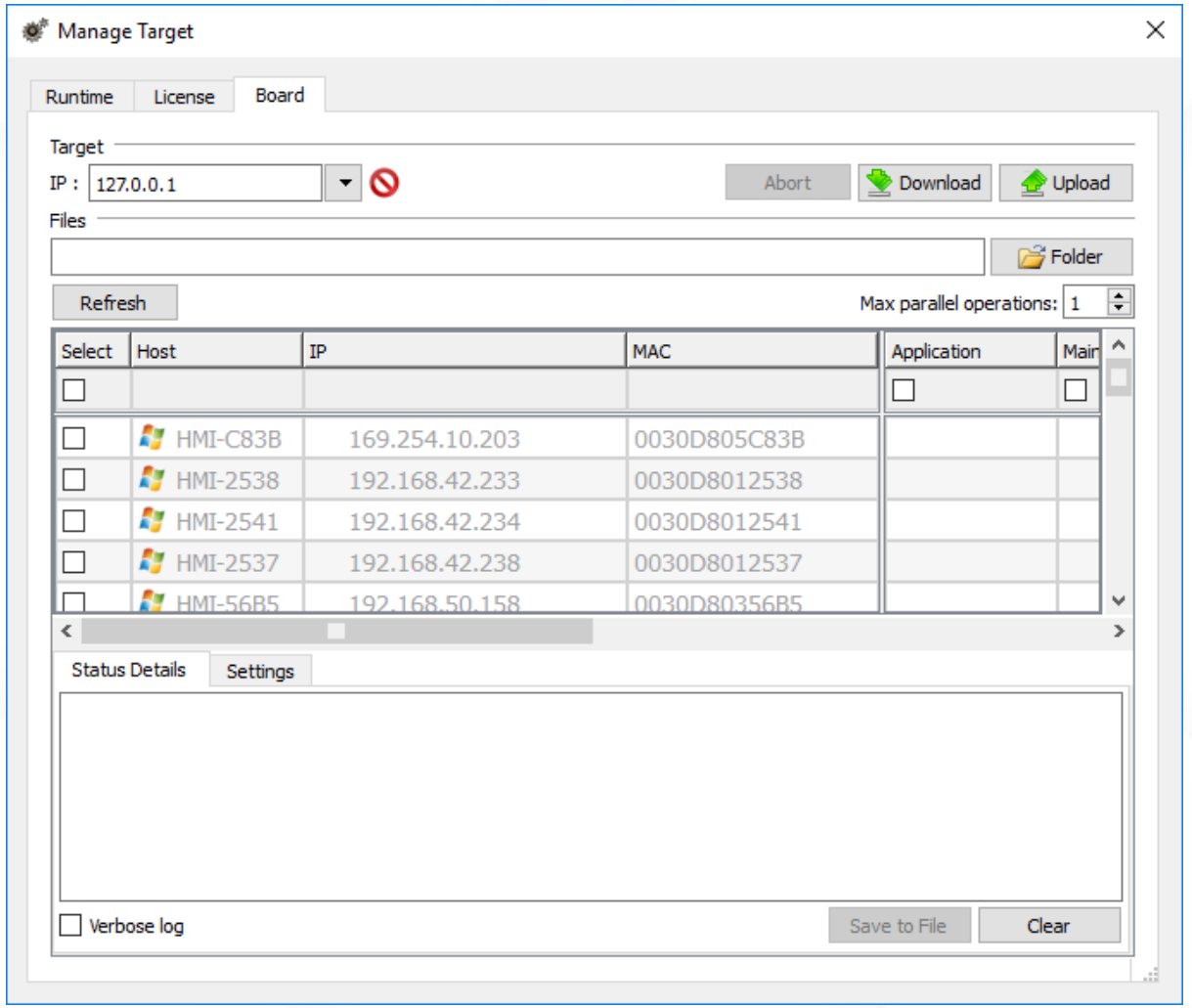
Manage Target - Board

Update OS components via Ethernet

Useful to retrieve OS components info

Useful to replace HMI splash screen shown during power-up phase

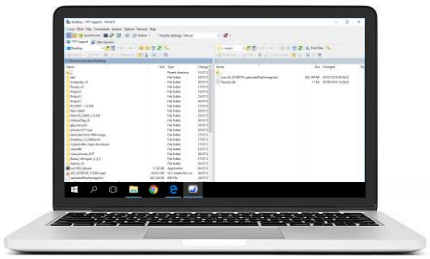
- type: 16bit bitmap
- filename: splash.bmp
- size: less than 500 KB
- resolution: even on width and height



Remote Control

Remote Control overview

FTP Client



download
upload

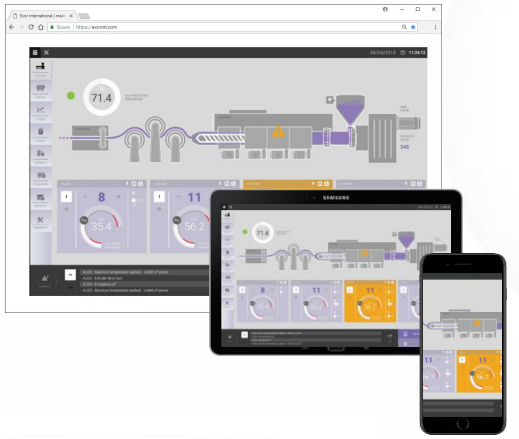
HMI
with JM Runtime and Project

JMobile Client

Native client

JM4Web

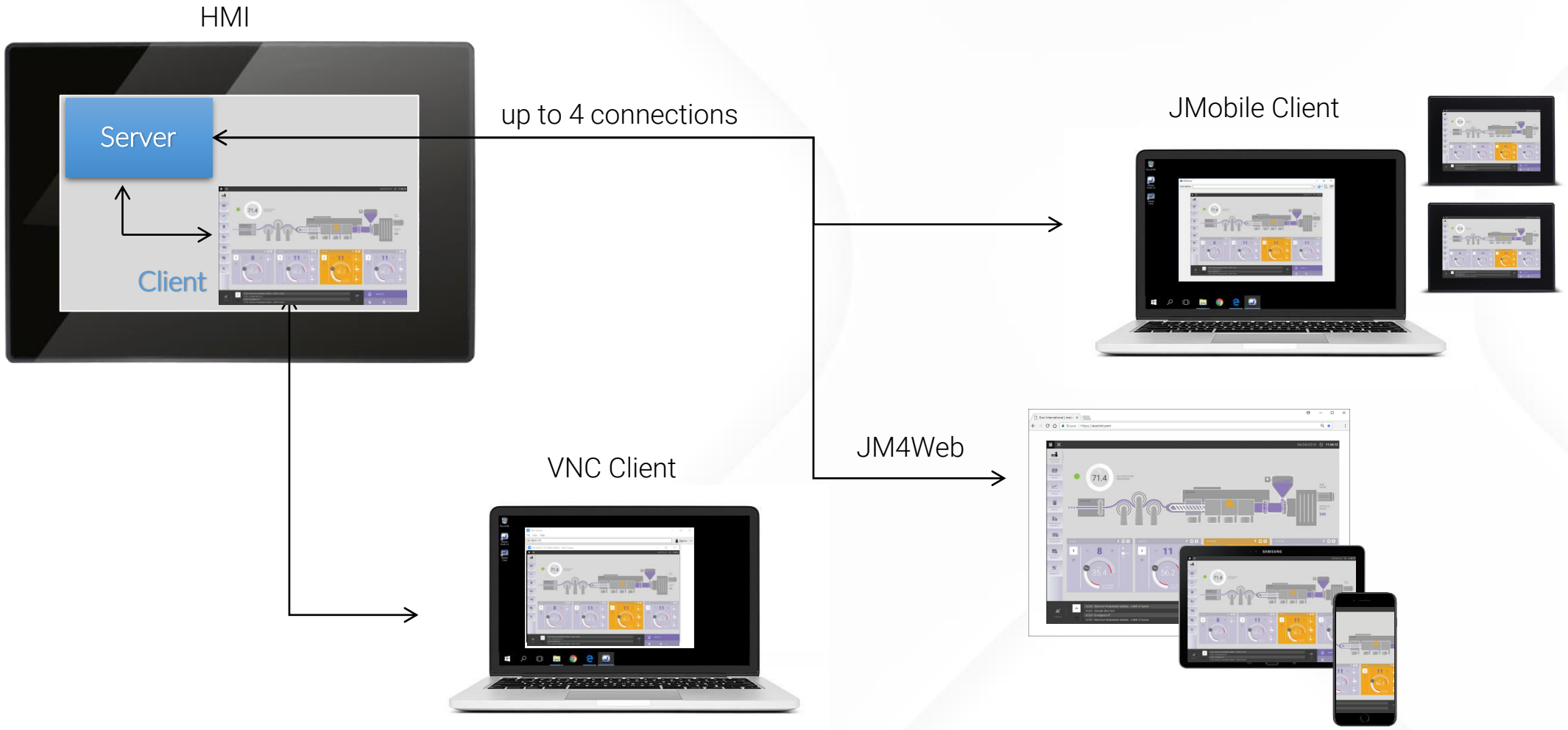
Http pages



VNC Client

Remote Desktop

Remote Control - Clients comparison



VNC

Any action executed on HMI screen is viewed in VNC Client and vice-versa

Sends graphic frames and click commands based on VNC standard protocol

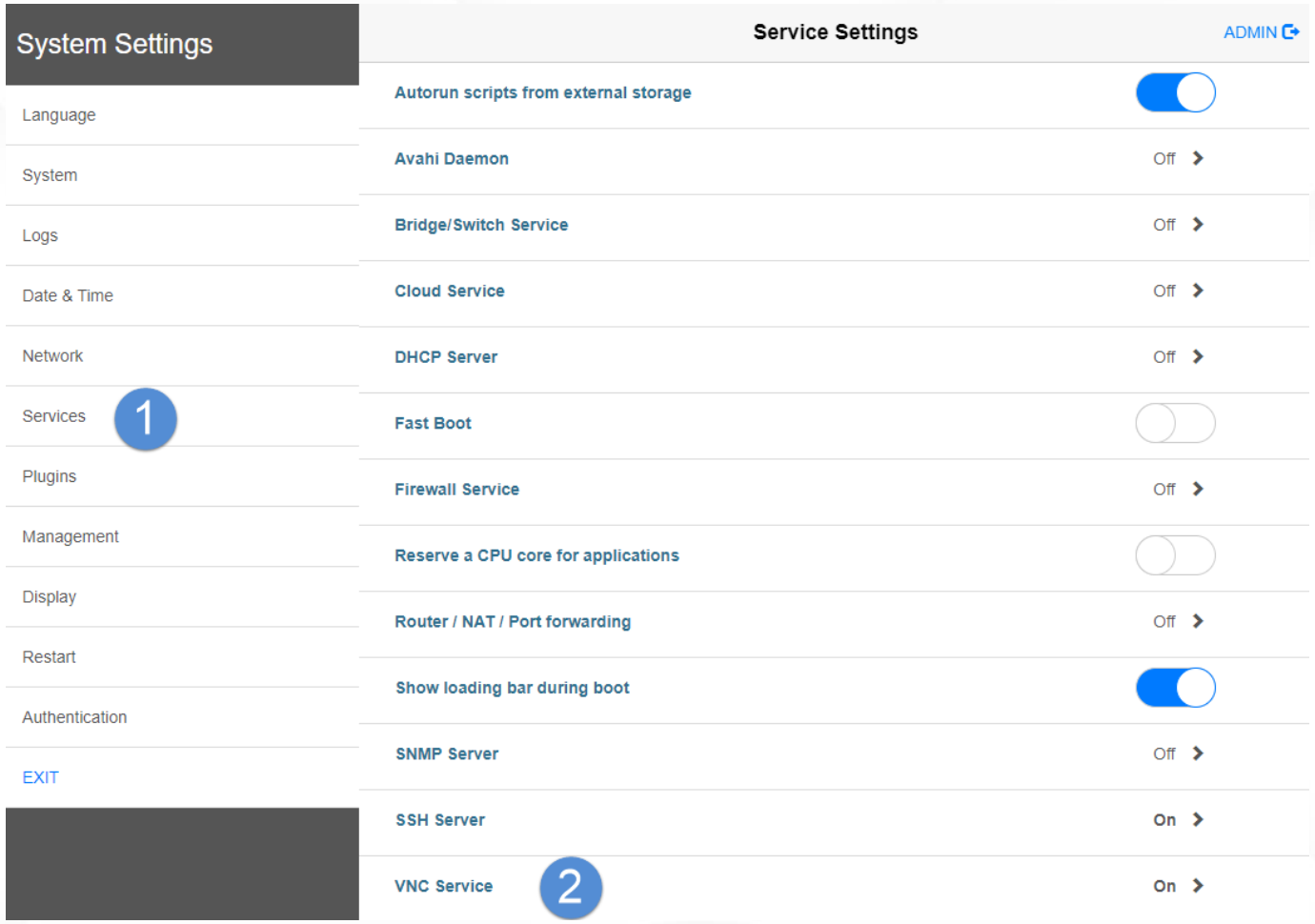
Useful to “teach” end user to operate on HMI application

Enable VNC

From System Settings

Context menu: directly on HMI

Browser:
https://<ipAddress>/machine_config
https://<ipAddress>/system_settings



JMobile Client

JMobile Client is an application included into JMobile Suite installer package

It runs on Microsoft Windows OS machines

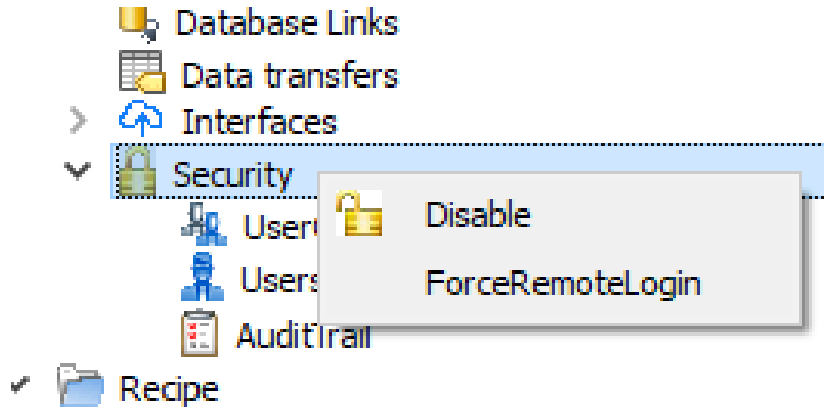
Same application running on the HMI will be loaded in JMobile Client

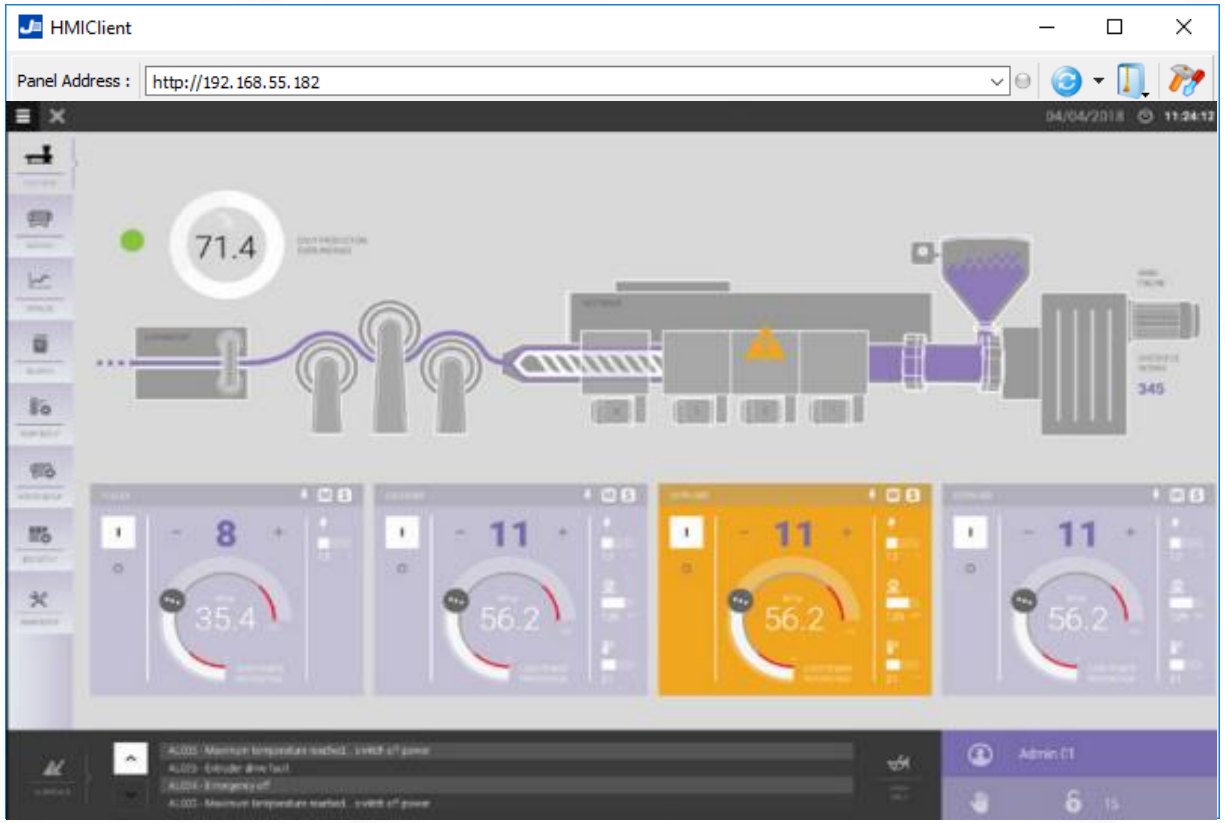
Page navigation on JMobile Client is independent from HMI

JMobile Client

Access from JMobile Client is secured by "Force Remote Login"

Once launched, you need only to type-in HMI address





JM4Web

JM4Web provides a real-time interface
to the HMIs using standard web technologies

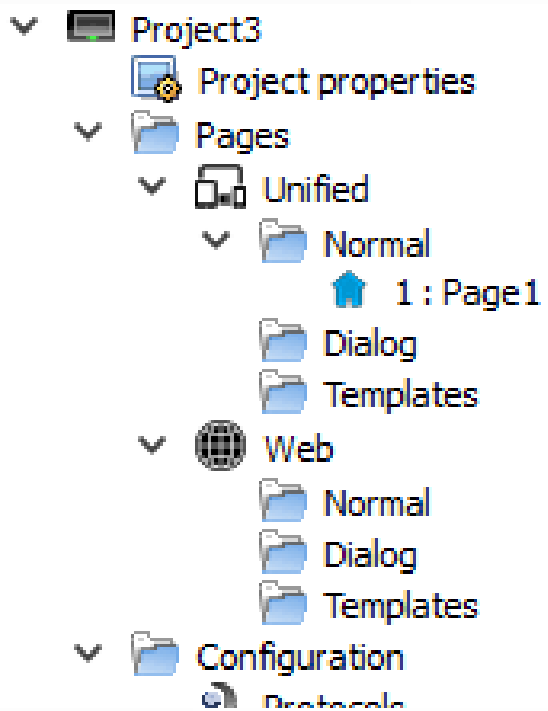
Runs on HTML5 web browsers

Web pages can be

common to native page (Unified)

separated (Web only)

NEW
in 4.0



JM4Web

NEW in 4.0

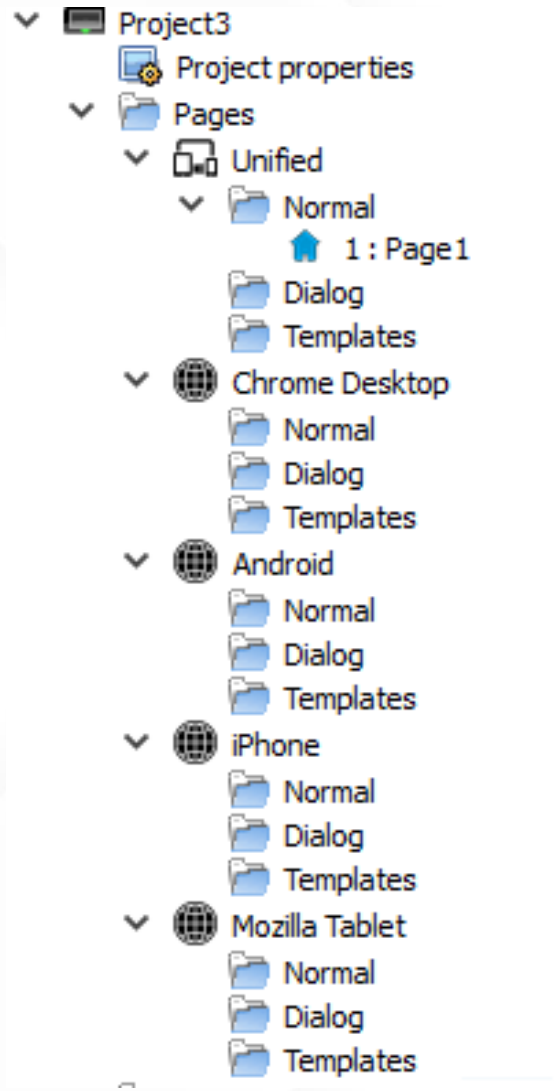
More than one Web technology profile into one project

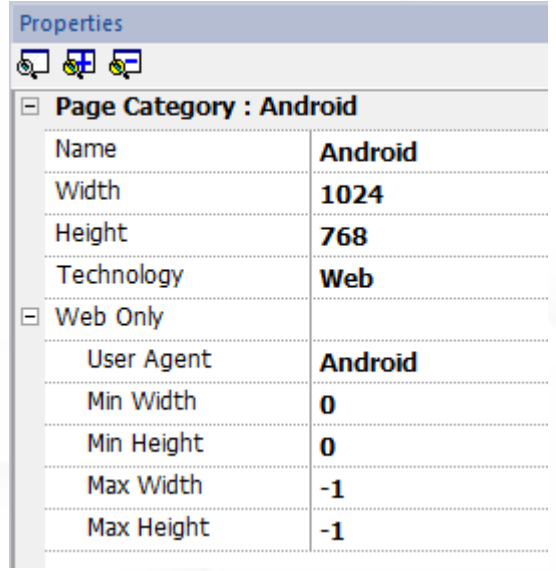
Pages can be different

based on device

based on “User Agent” parameter

Pages can be shared between categories



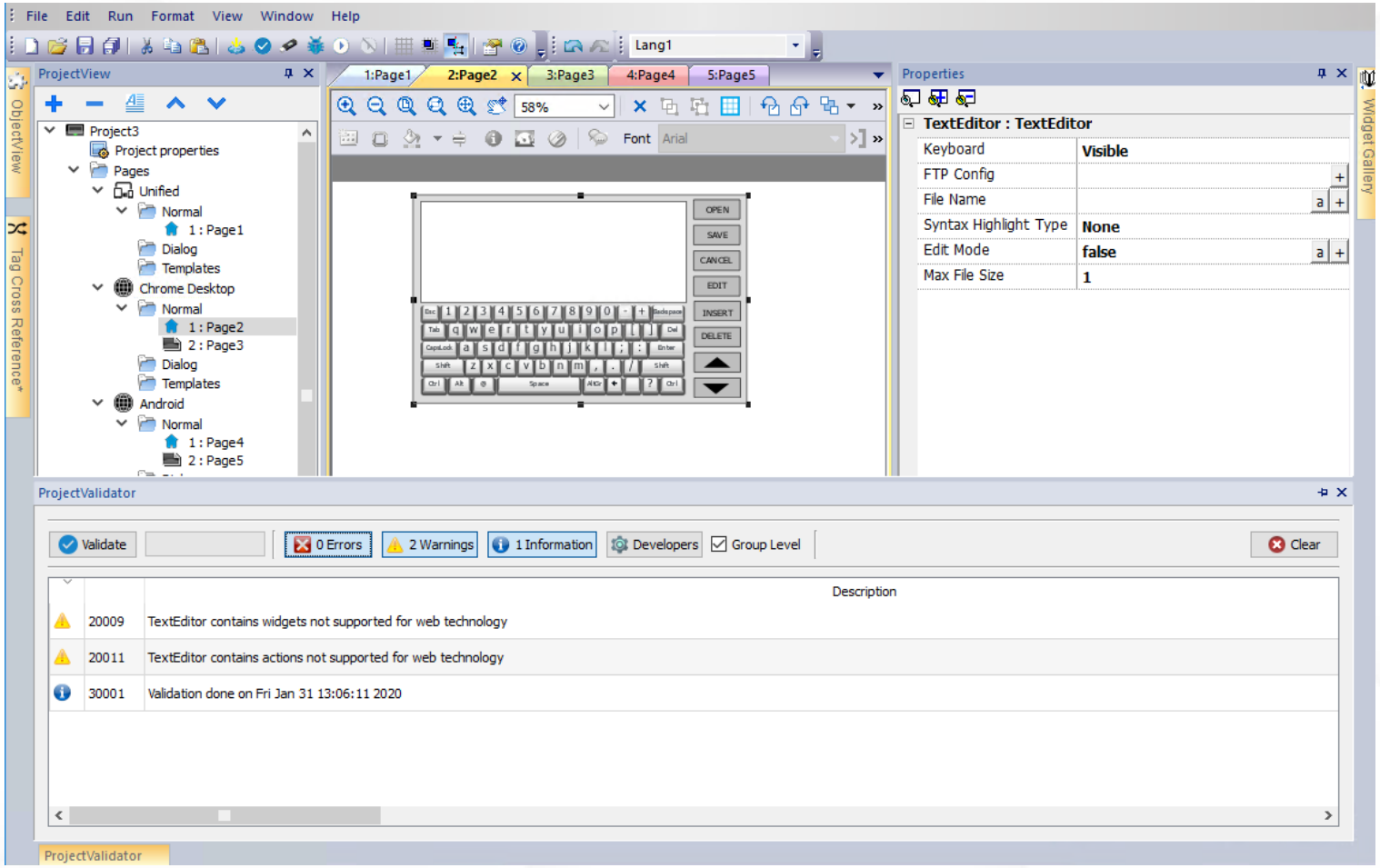


JM4Web – Project Validator

NEW in 4.0

Permits to check if widgets or actions are supported into web pages

All adjustment can be done by user before downloading project to target



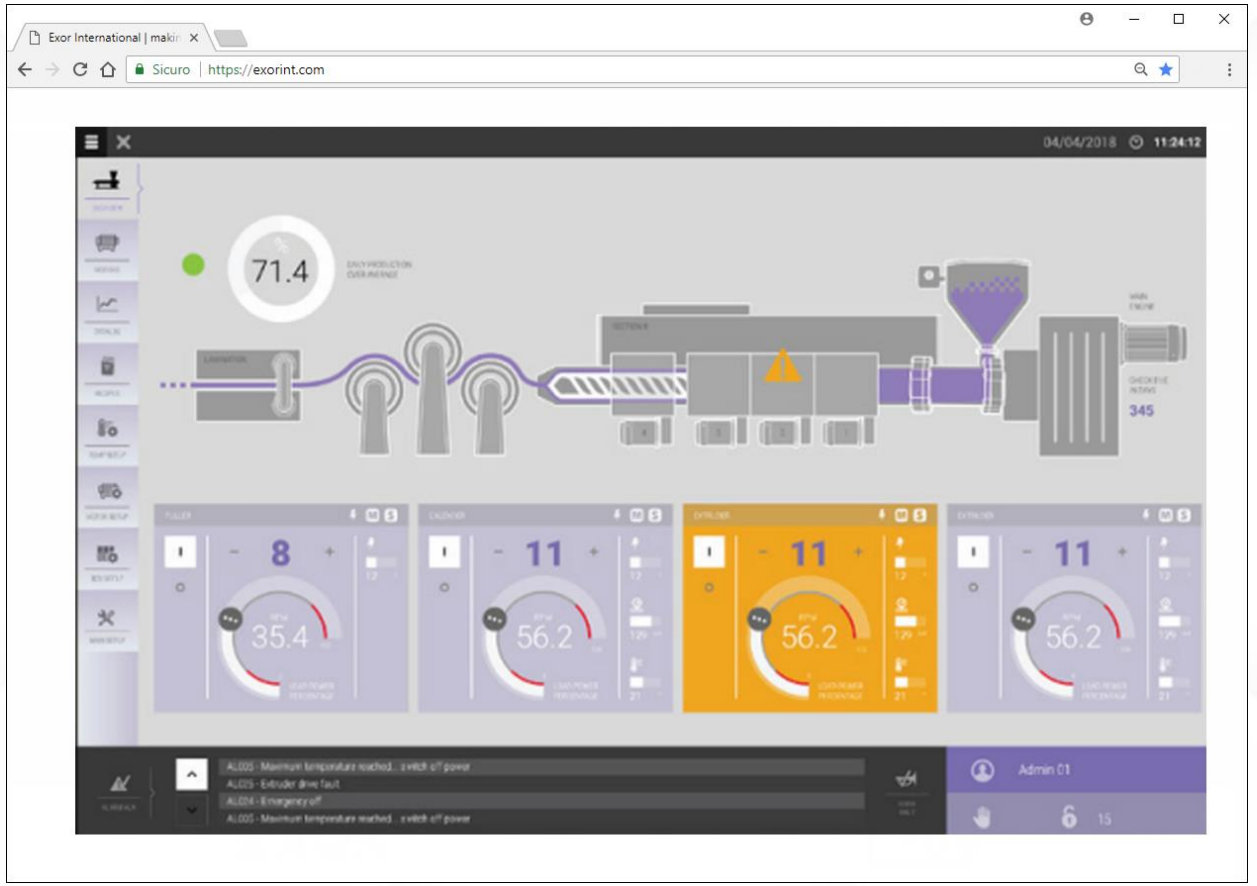
JM4Web

After project download, web page are reachable from browser, pointing to

http://<IPAddress>

If HMI is not available, web pages can be reached also in Simulation

http://<IPAddress>:81



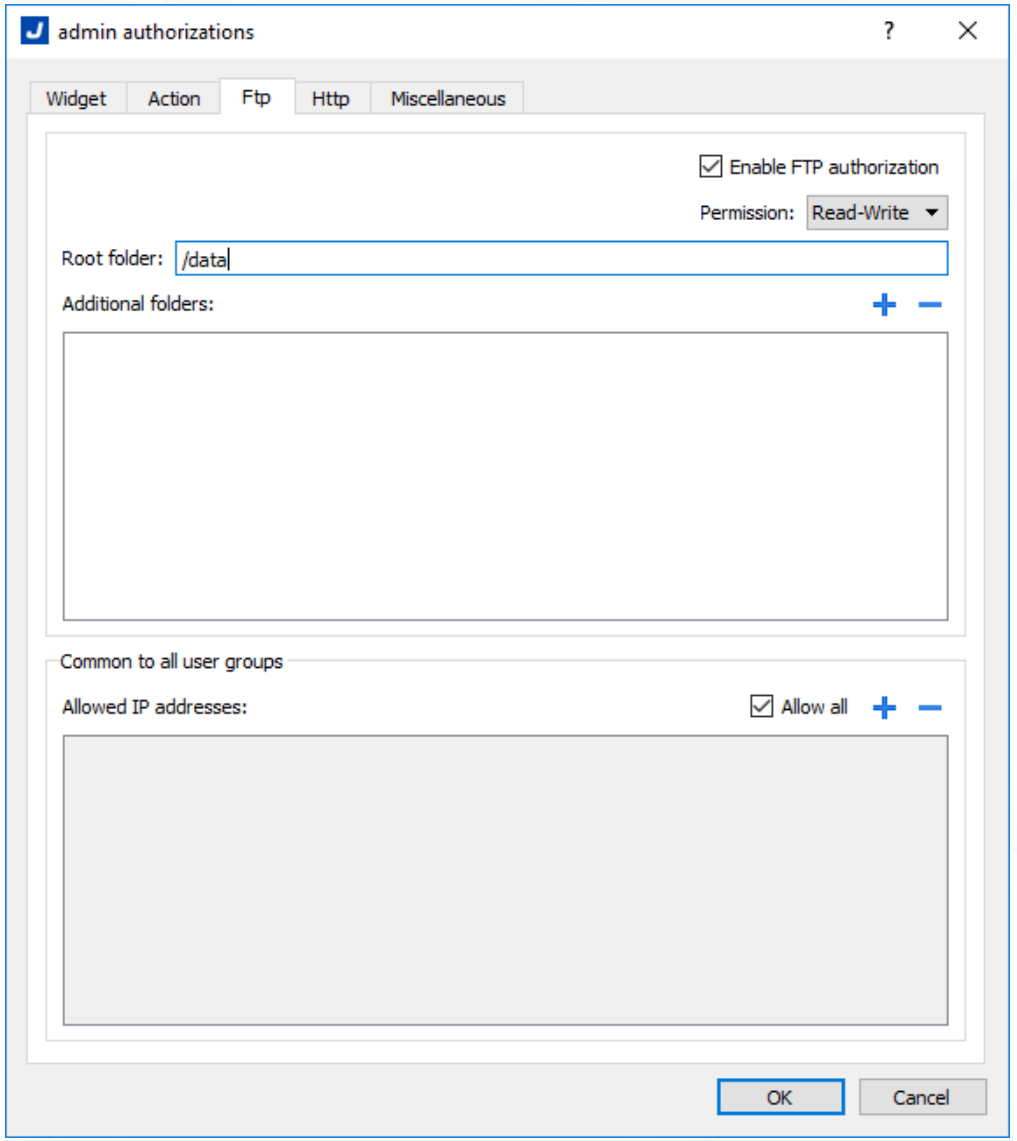
FTP

Enable FTP Server on HMI
to get access to the internal flash disk data

Any standard FTP Client can be used

FTP access is disabled by default and must be enabled into Group Authorization settings:

- "Enable FTP authorization"
- "Root Folder" can be set as per needs

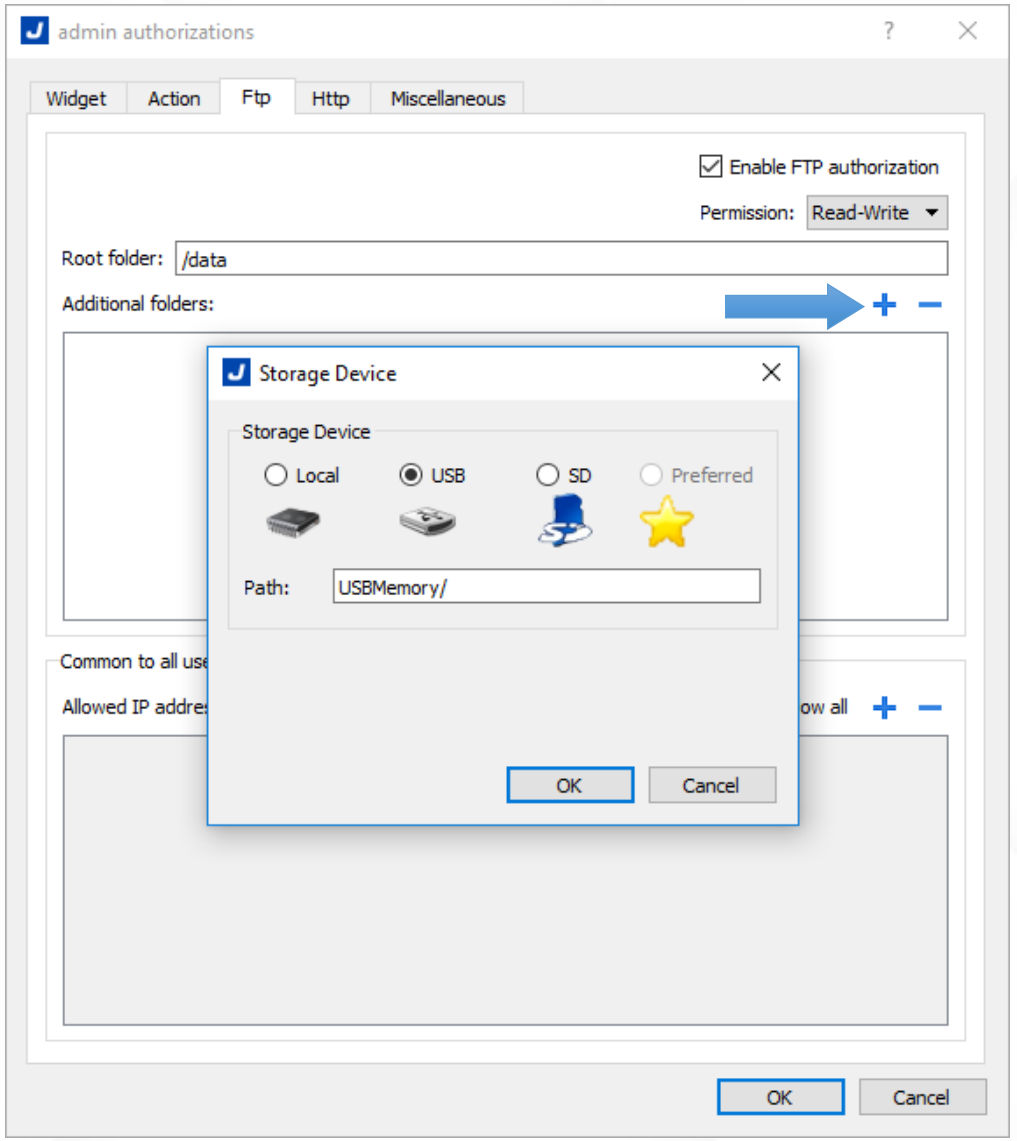


FTP access to external memory

Possibility to map “USB Memory” or “Storage Card” as FTP Folder

USB Memory and SD Card will become accessible by FTP connection

Very useful to retrieve Alarm, Trend or Recipe files dumped into external memories

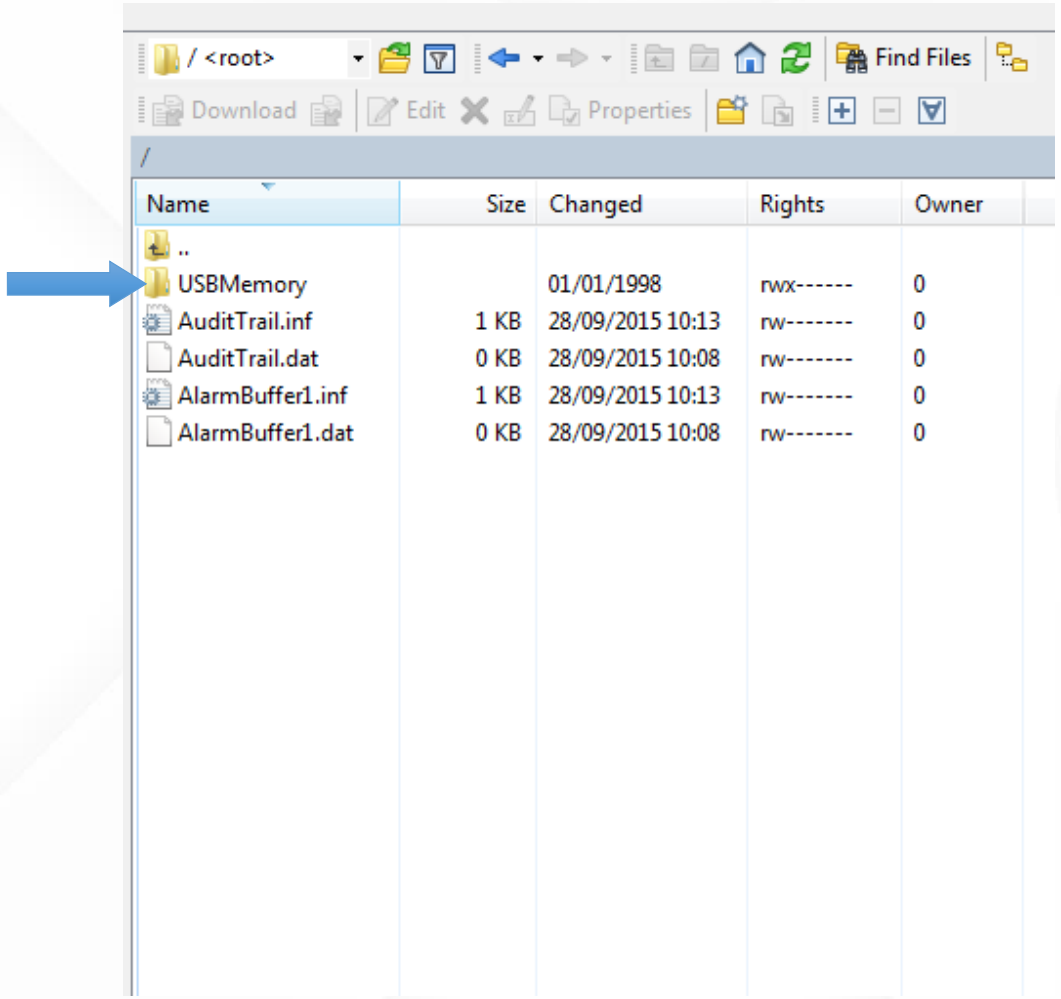


FTP access to external memory

Possibility to map “USB Memory” or “Storage Card” as FTP Folder

USB Memory and SD Card will become accessible by FTP connection

Very useful to retrieve Alarm, Trend or Recipe files dumped into external memories



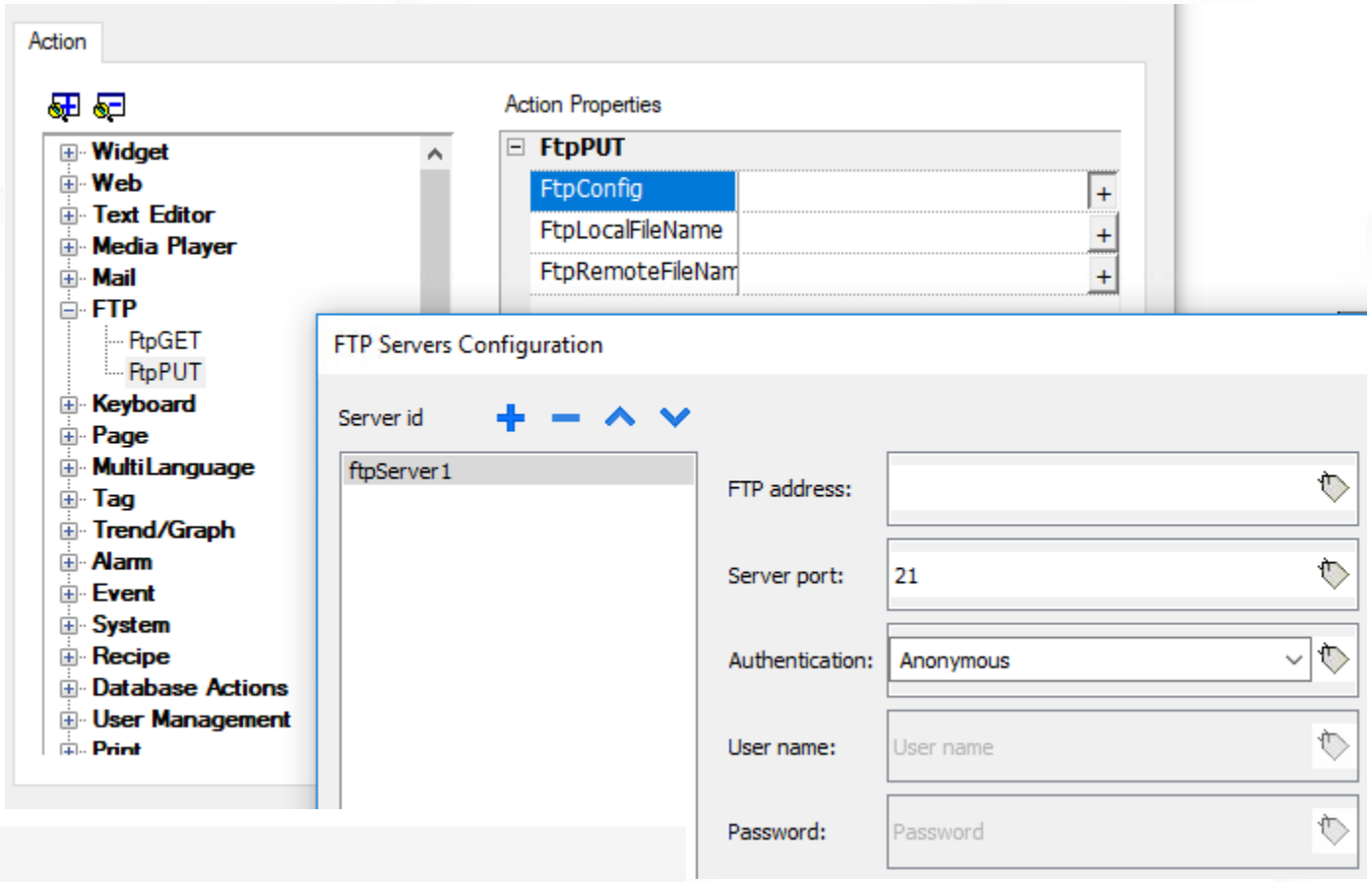
FTP client

Action to access to an FTP server

- FTP Put
- FTP Get

From native actions

From JavaScript



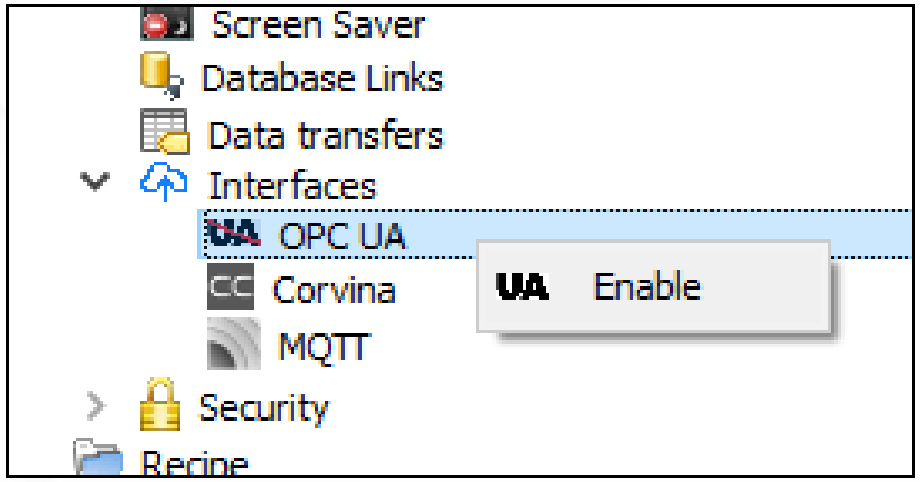

```
project.ftpCONFIG("192.168.0.200", "21", "true", "admin", "admin");

project.ftpGET( "data.txt",
                "\\USBMemory\\data.txt",
                function(ftpStatus) {fnFtpGetFinished(ftpStatus);} );

function fnFtpGetFinished(ftpStatus) {
        alert(ftpStatus);
}
```

Interfaces

OPC UA Server

All configured Tags can be exposed as OPC UA Server variables

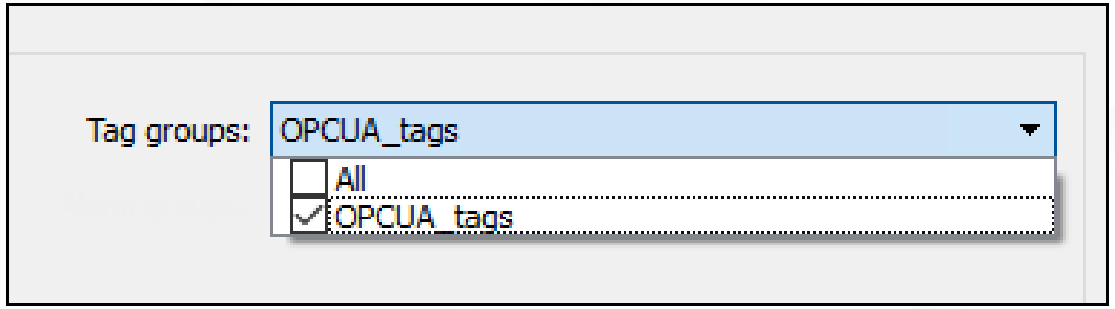
Accessible from any client based on OPC UA specifications



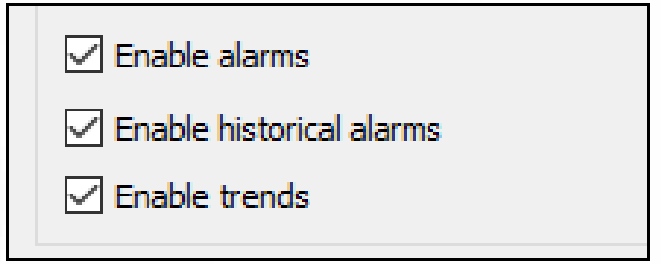
Tip: can be simulated launching Simulator (no necessary need of target device)

OPC UA Server

Possibility to choose all Tags or specific group

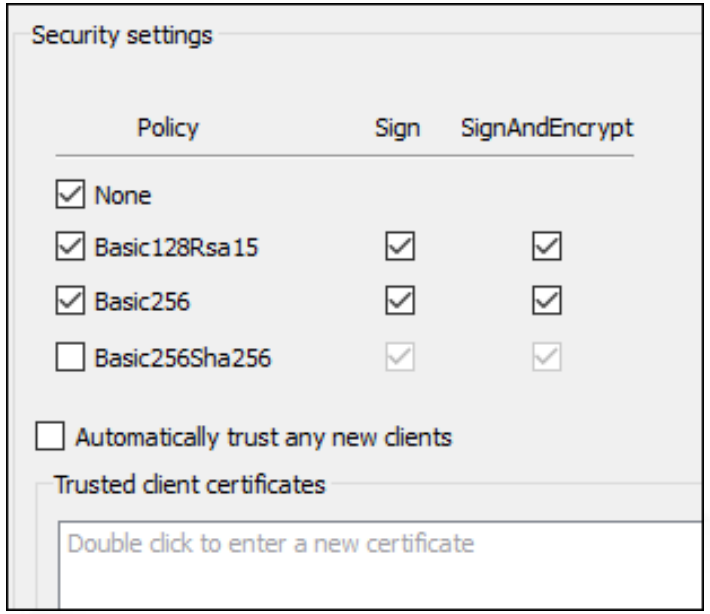


Expose active/historical alarms and trends



Allowed connections based

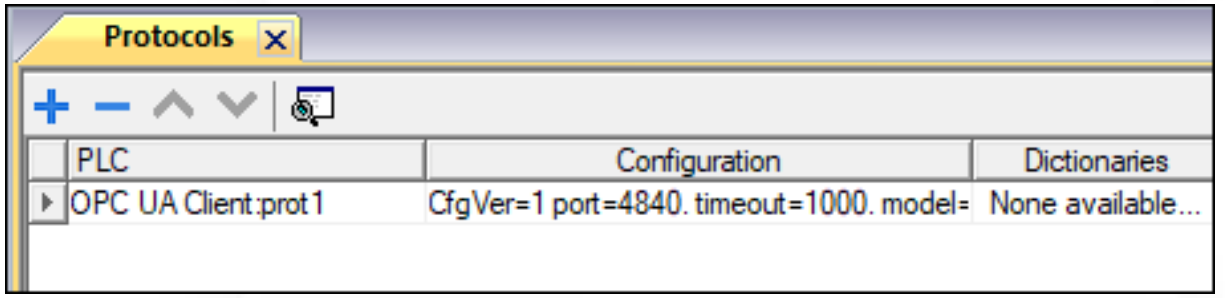
None: typical for communication only
Sign: uses credentials
Encrypt: uses certificates



OPC UA Client

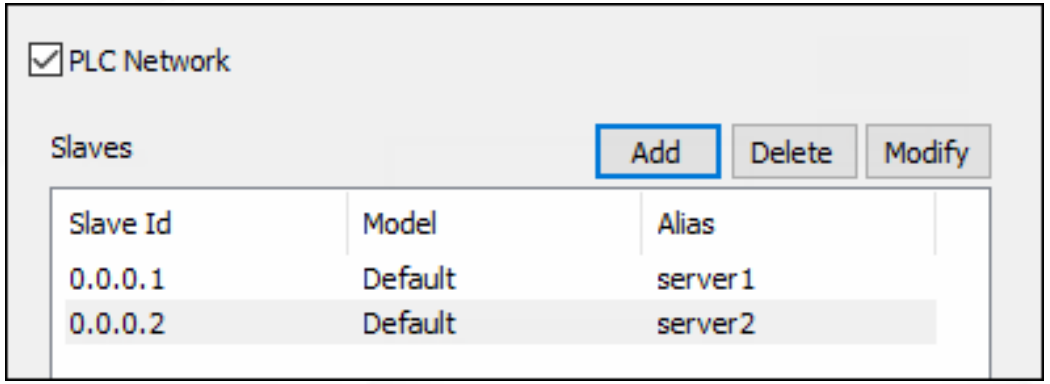
Can access to any server based on OPC UA specifications

Selectable from protocol list



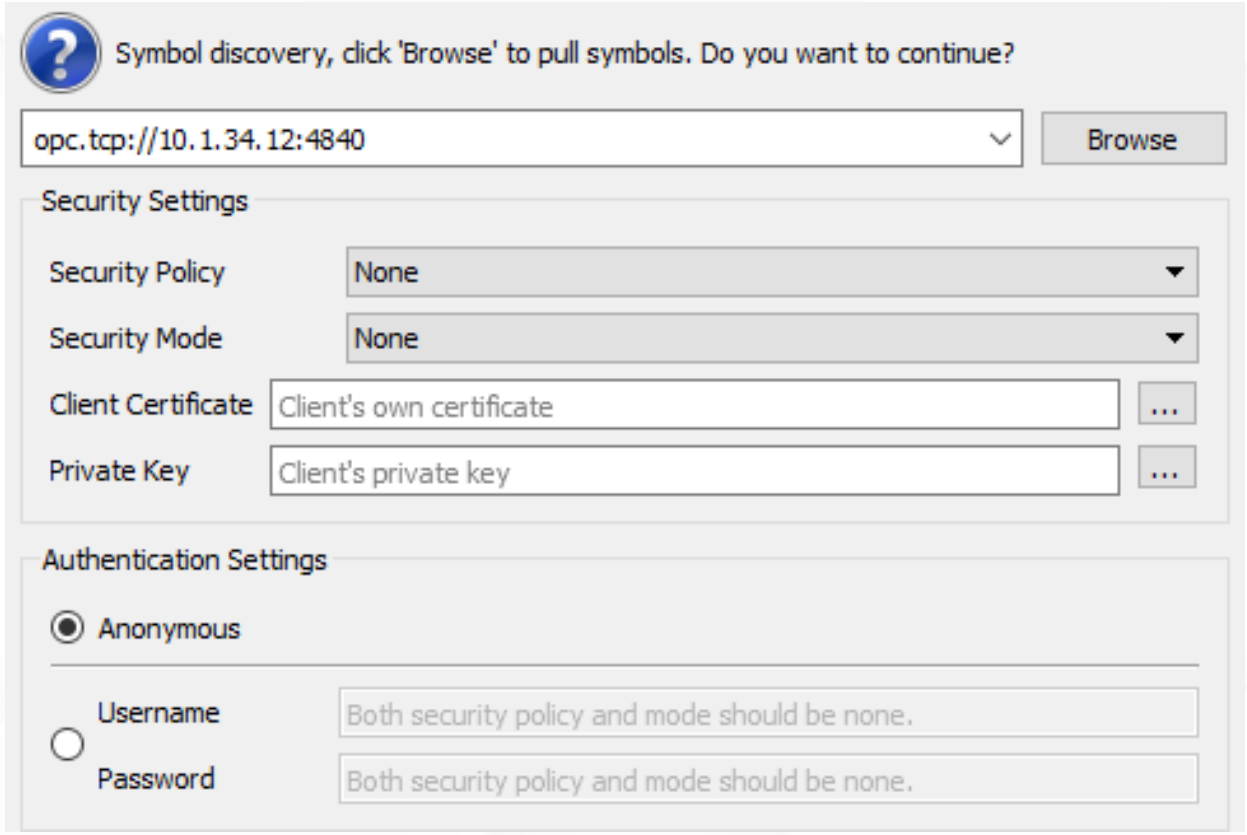
Supports PLC Network

One protocol to connect to many OPC UA Server



OPC UA Client

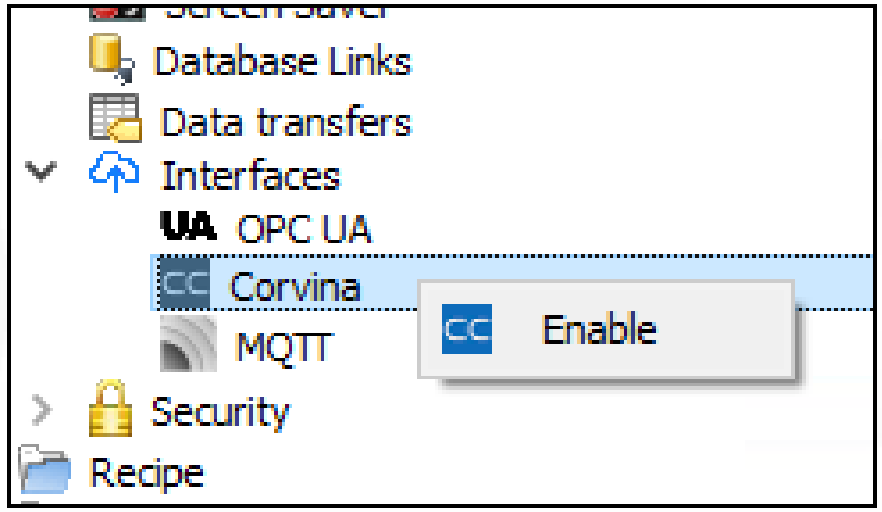
Online discovery of OPC UA symbols exposed from OPC UA Server by using Tag Importer



NEW
in 4.0

Corvina

Connector for Corvina Cloud * portal
for data pushing to cloud



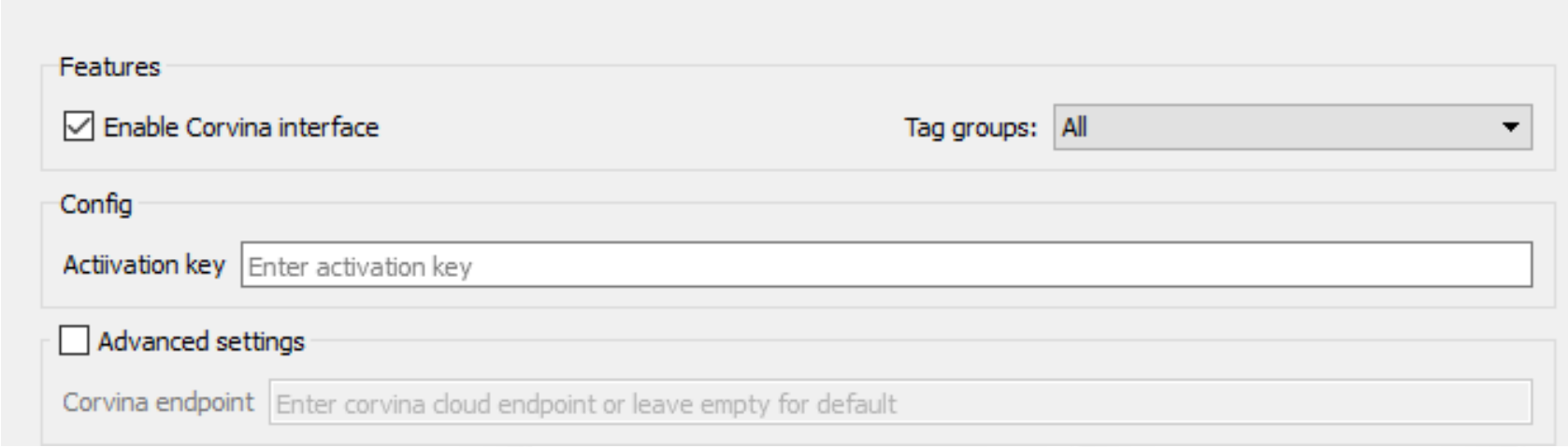
* Corvina Cloud phase 2: data pushing to cloud

NEW
in 4.0

Corvina

Select Tag group to be pushed to Corvina Cloud

Enter Activation key to couple project with device defined on Corvina Cloud portal



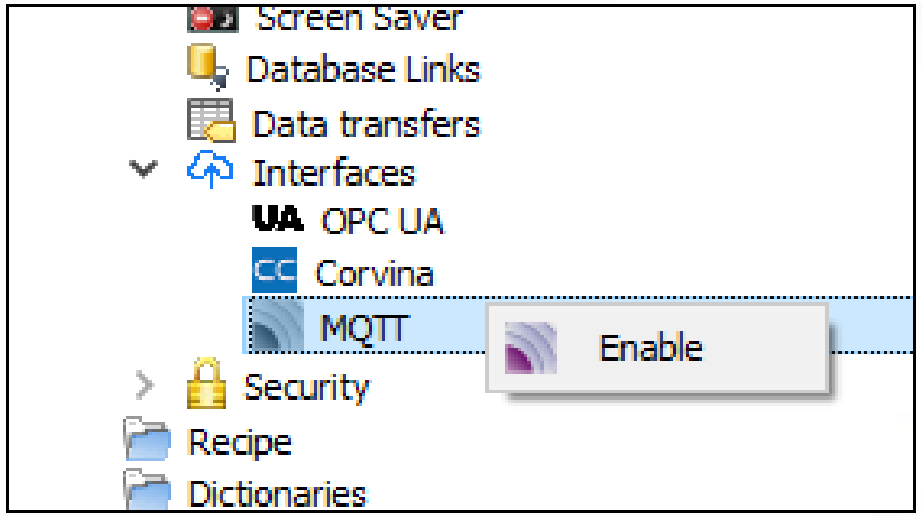
NEW
in 4.0

MQTT

Connector for MQTT broker to push data into cloud

Presets available

- Azure
- Amazon AWS
- IBM BlueMix
- Murano
- Generic MQTT broker

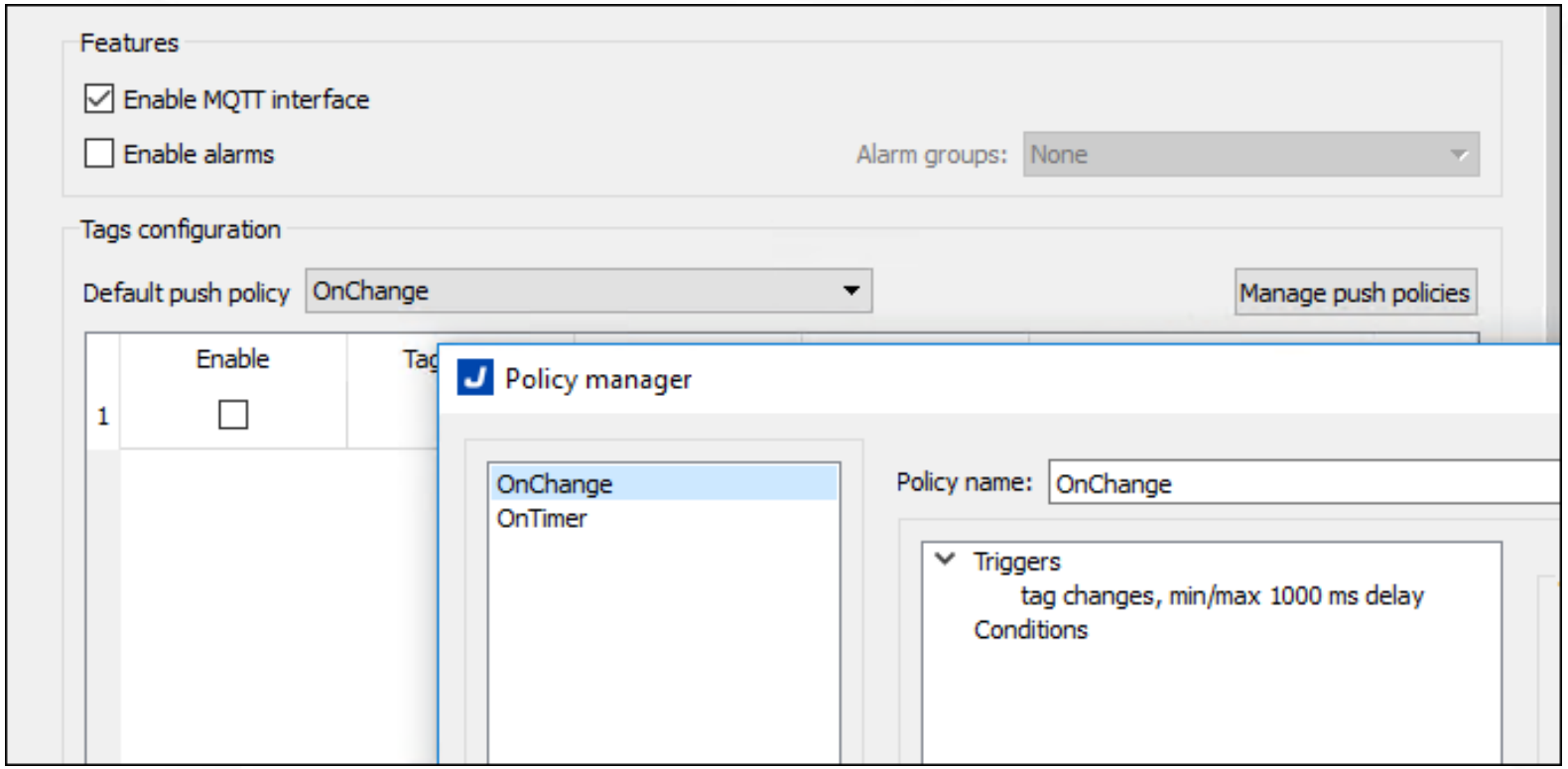


NEW
in 4.0

MQTT

Set rules for data pushing

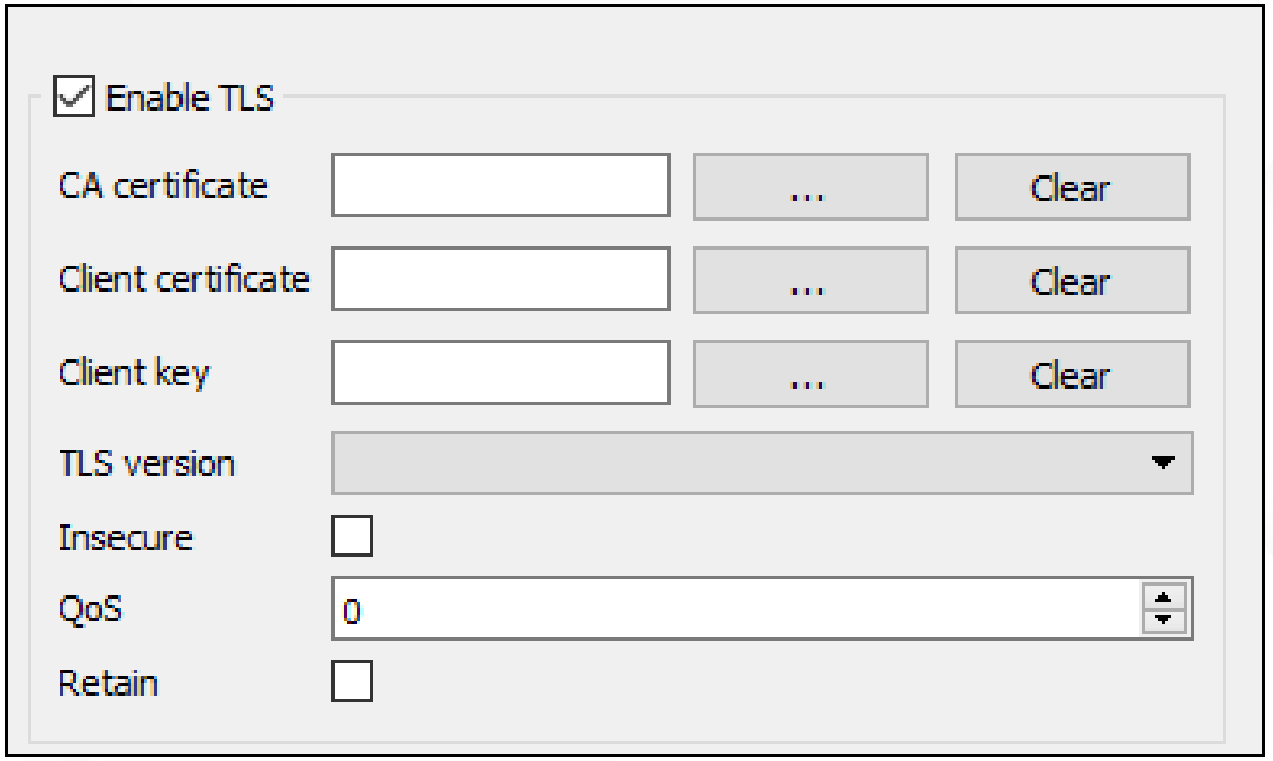
- Based on timer
- Based on value change



MQTT

NEW
in 4.0

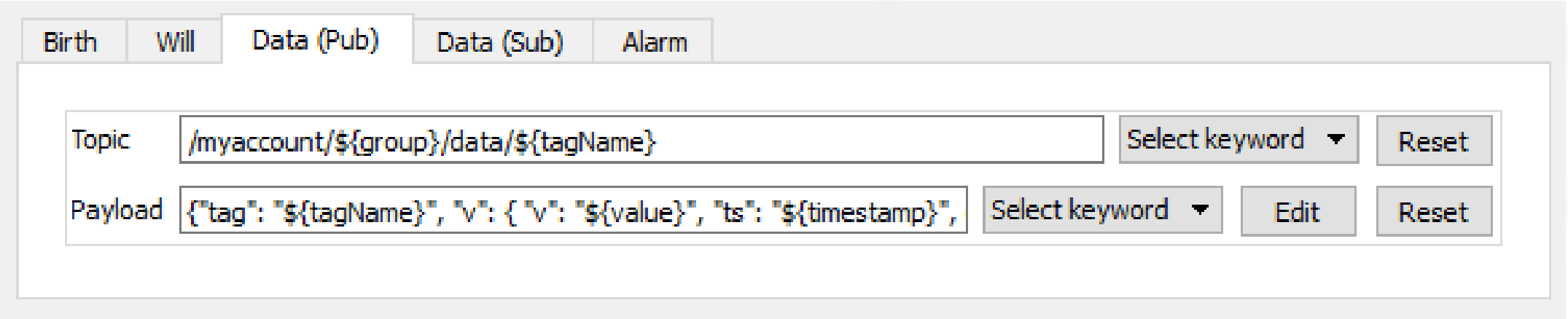
Enable security options



NEW
in 4.0

MQTT

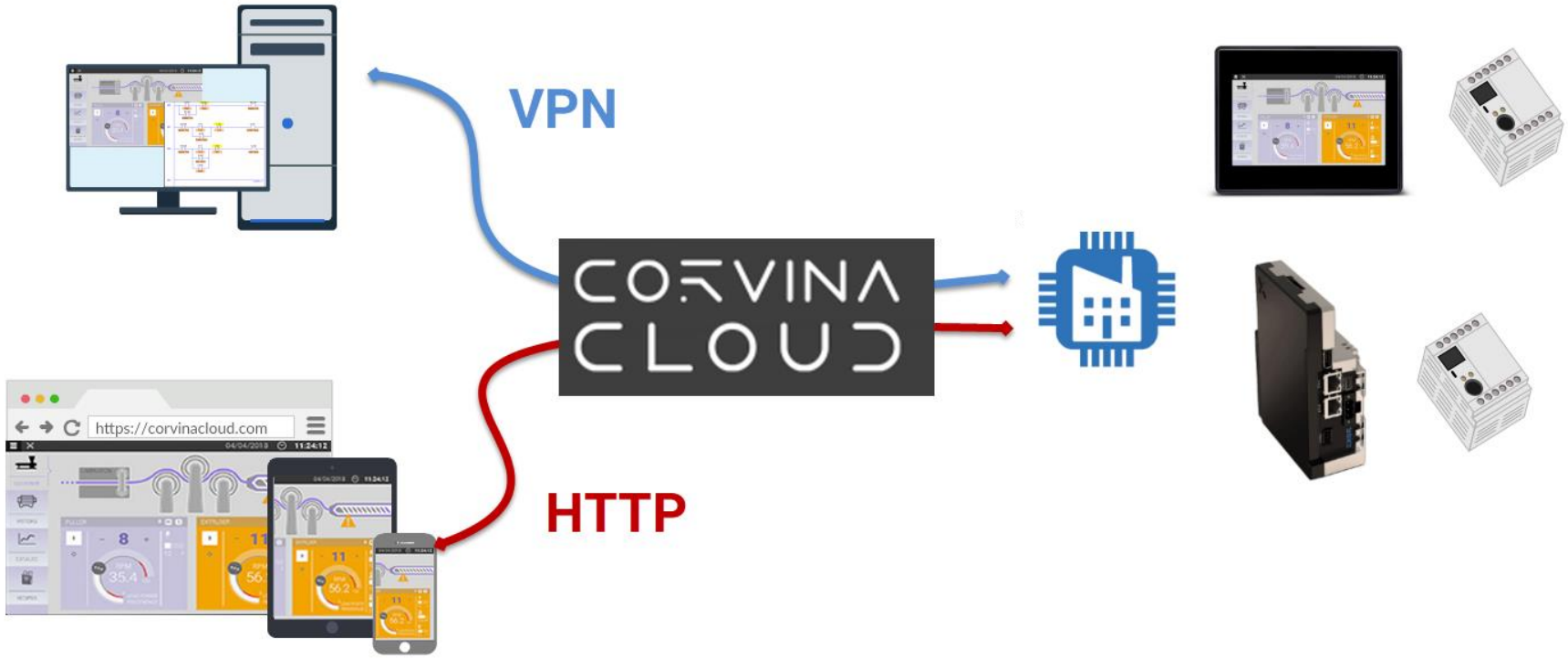
Create customize data message format



Corvina Cloud

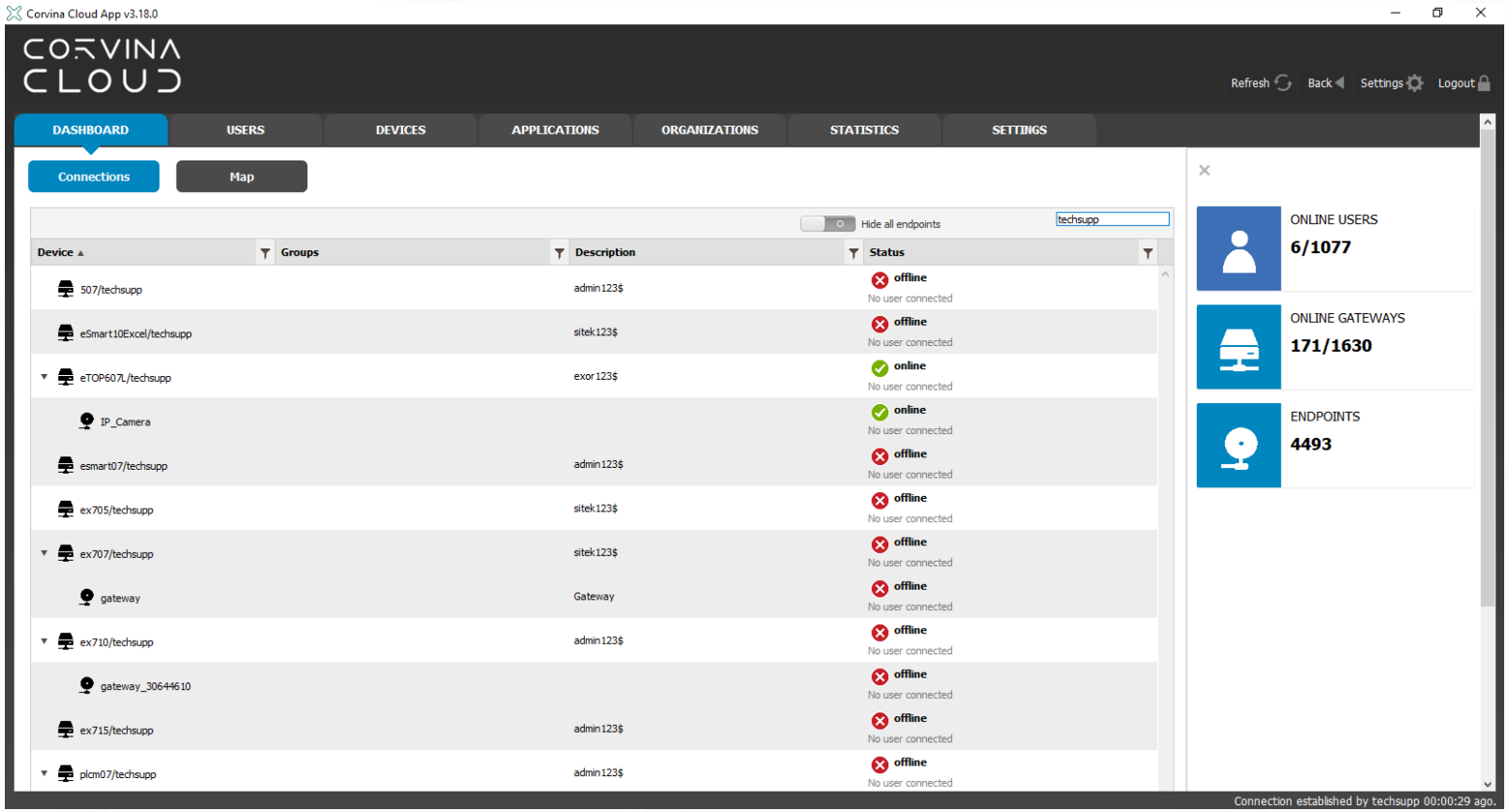
What is Corvina Cloud

Corvina Cloud is the EXOR International solution for
Remote Control and Teleassistance, based on secure VPN and Web access



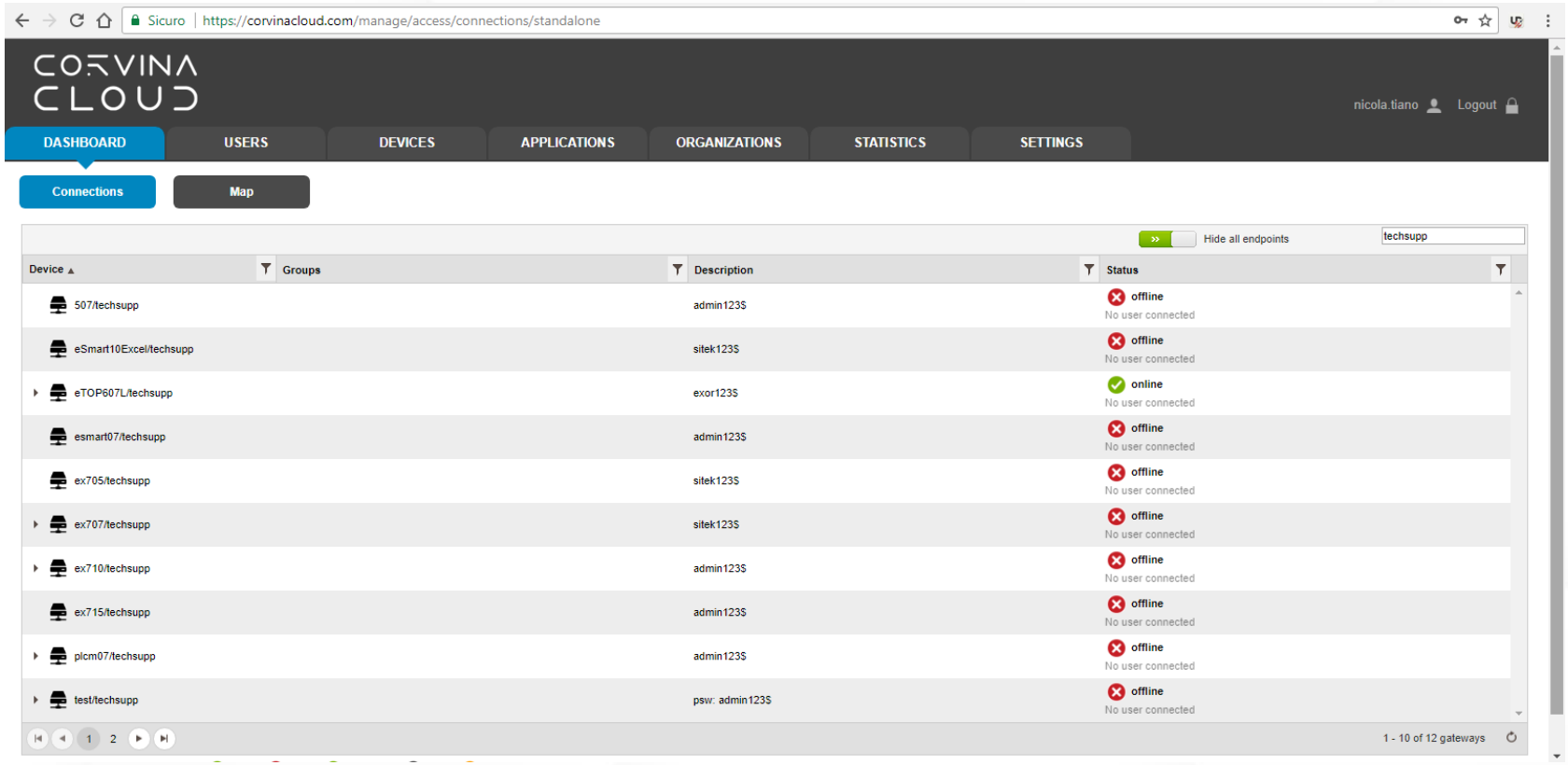
Corvina Cloud App

VPN can be opened using Corvina Cloud App



Web Access

Web access is available connecting to corvinacloud.com



Why using Corvina Cloud

Reduces costs of maintenance and support

Provides connectivity for new or existing plants

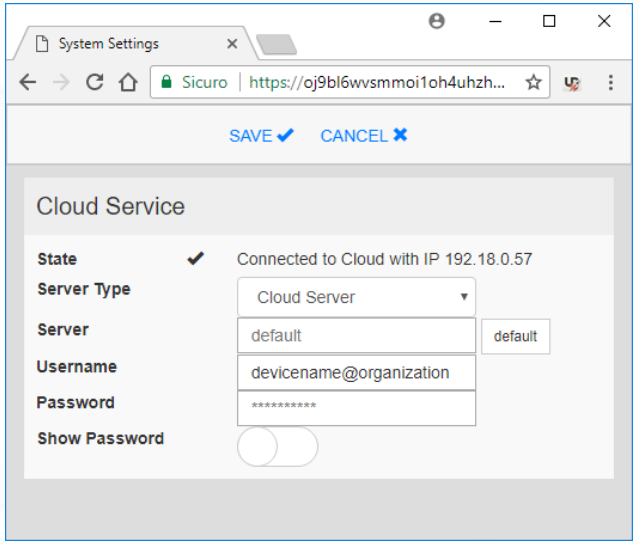
Provided by same HMI manufacturer (native solution)

Completely accessible and configurable from browser

Corvina Cloud solutions – JSmart700 series

One can be dedicated for remote control, others can be logically separated and dedicated to communication with PLC

Cloud service configurable from web browser

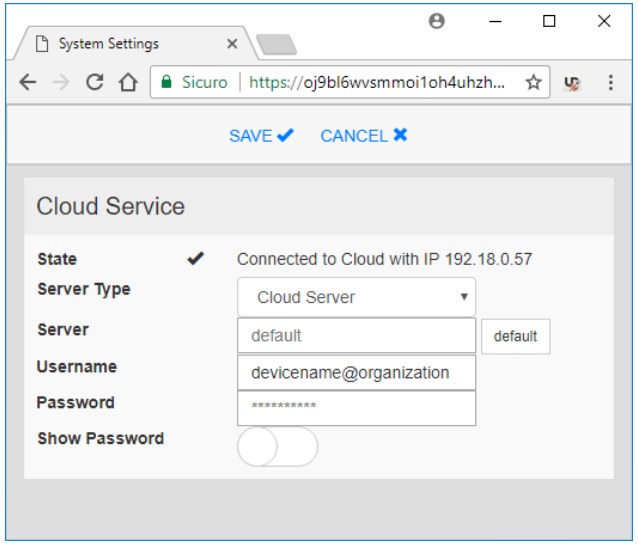


Corvina Cloud solutions – eX700/eXware700 series

Separated Ethernet interfaces

One can be dedicated for remote control, others can be logically separated and dedicated to communication with PLC

Cloud service configurable from web browser

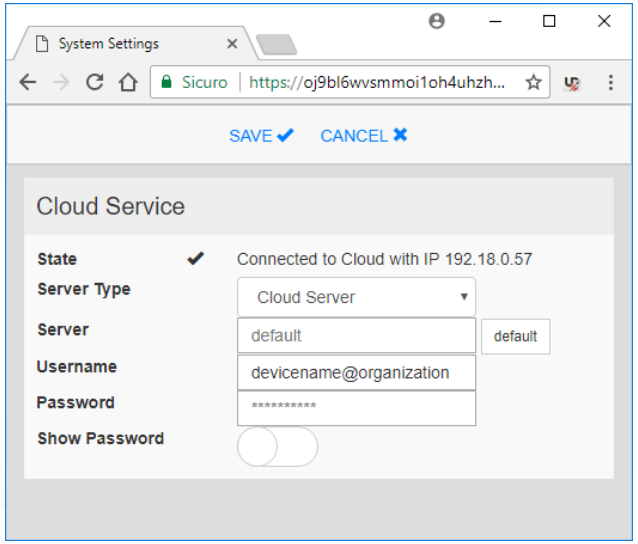


Corvina Cloud solutions – eSMART series

One single Ethernet interface*

Can be dedicated for remote control, and communication with PLC

Cloud service configurable from web browser



* Valid for eSMART04/07M/10 while eSMART107 has 2 Ethernet interfaces

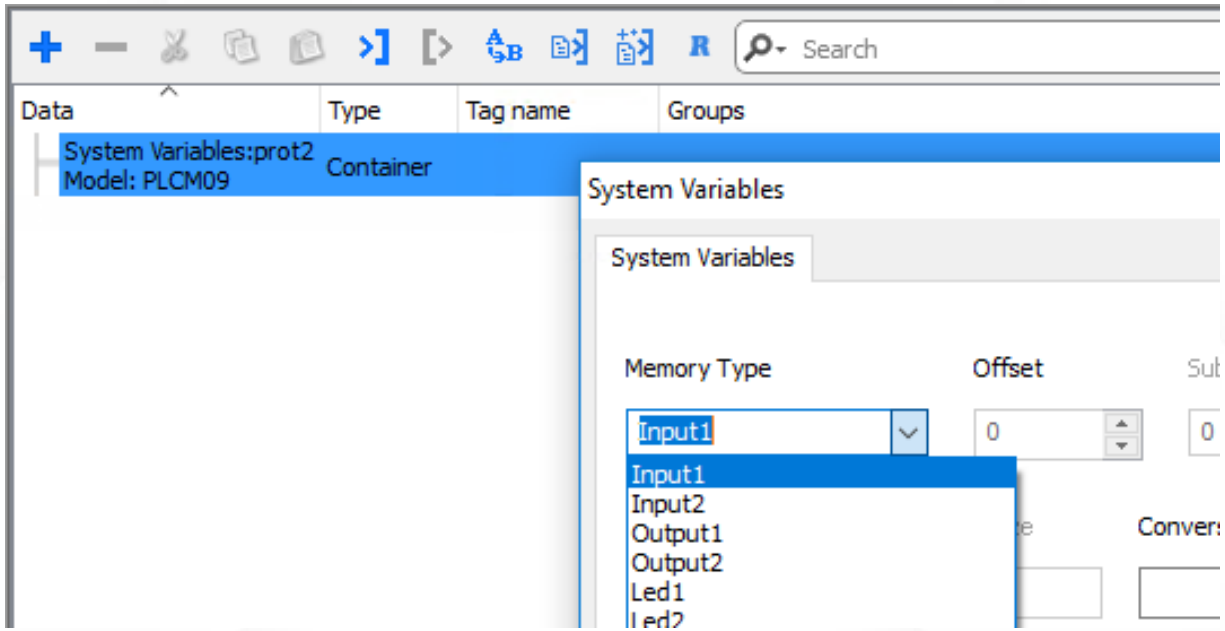
3G Modem module – PLCM09

Available for eX700 and eXware700 series

APN and SIM pin configurable from HMI System Settings

Configurable input and output
2 digital inputs/2 digital outputs (SSR) with optical isolation

JMobile System Variables available

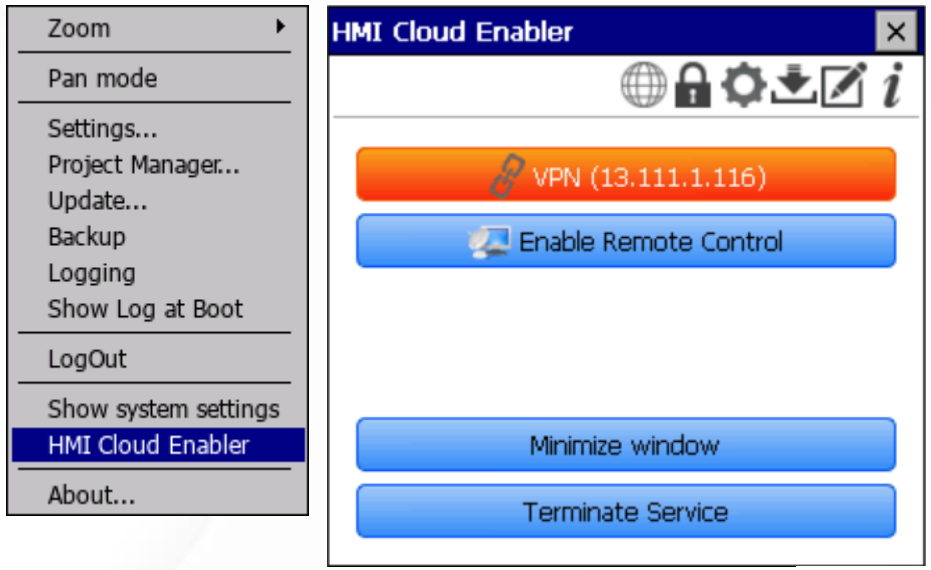


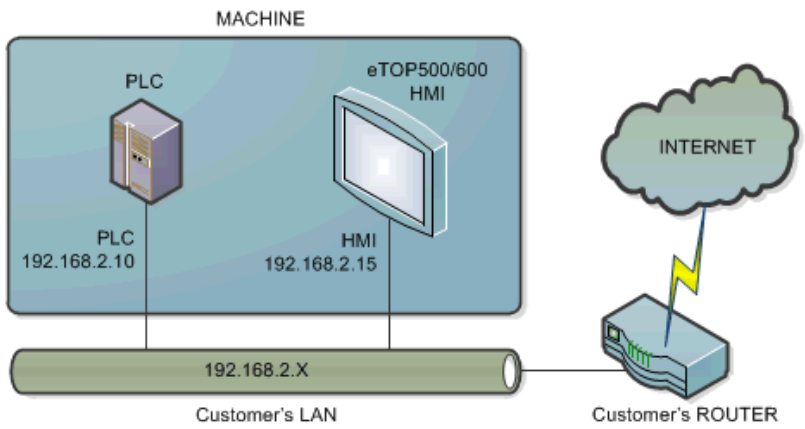
Corvina Cloud solutions – eTOP500/600 series

One single Ethernet interface
2 Ethernet ports

Can be dedicated for remote control, and communication with PLC using **HMI Cloud Enabler**

Configurable from HMI





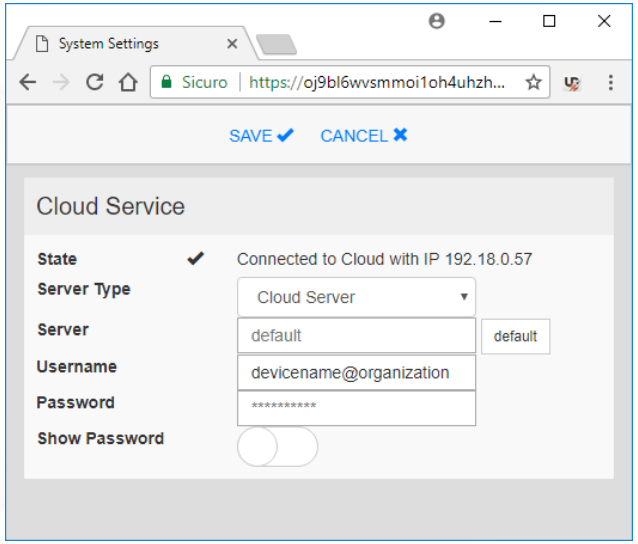
Corvina Cloud solutions – eTOP500/600 and PLCM07

Secure Cloud Connector with
2 separated Ethernet interfaces

Allows connection to Corvina service for eTOP500/600 panels when network separation is required

Cloud service configurable from web browser

Can be mounted on existing plants to give connectivity to eTOP500/600 HMIs

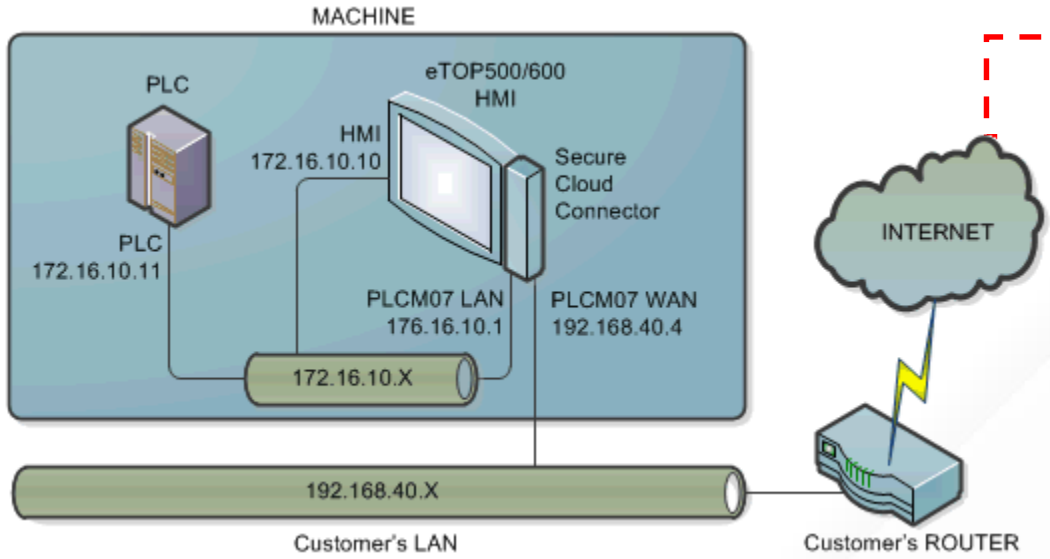


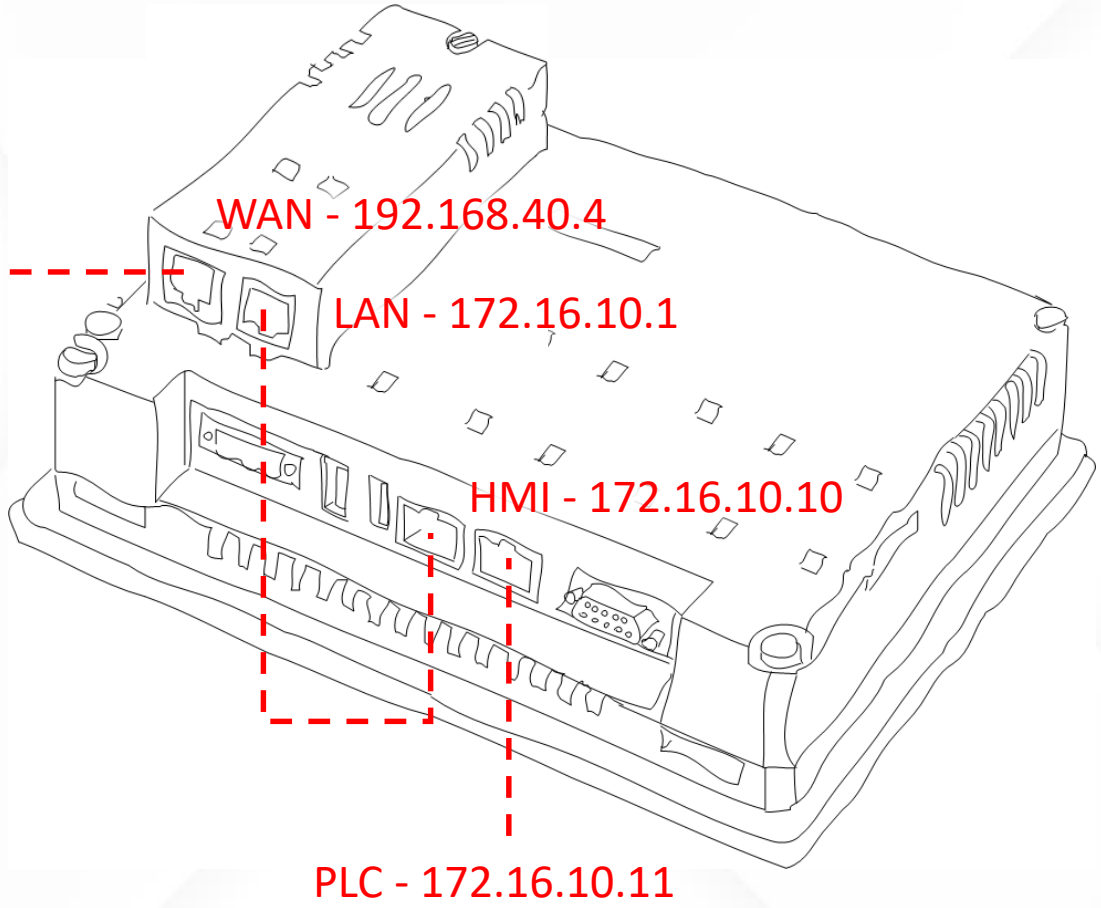
Corvina Cloud solutions – eTOP500/600 and PLCM07

Separate Networks

WAN = Internet / Corvina Cloud

LAN = HMI / PLC local network





Access to Corvina Cloud

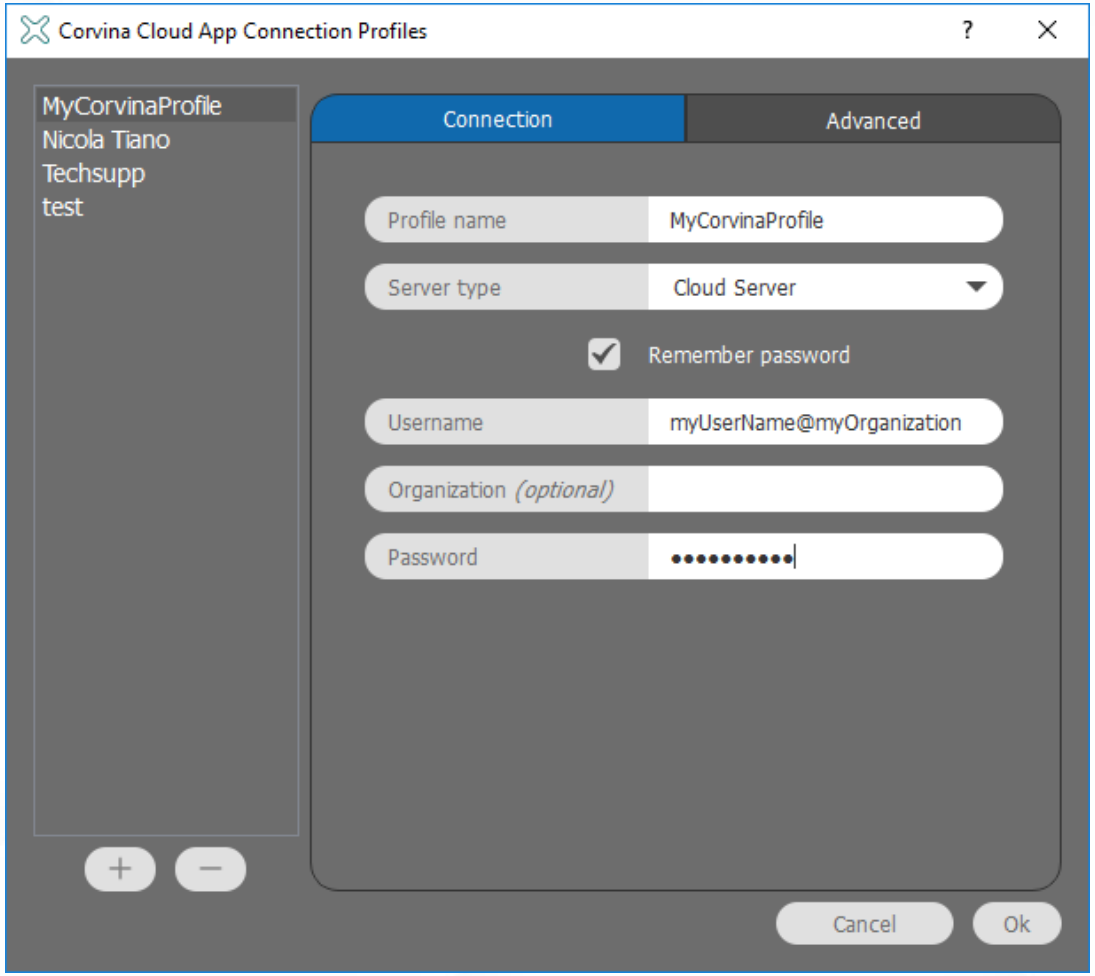
Access credentials to Corvina Cloud service to be requested to your EXOR products reseller

Access to Corvina Cloud

Username and password to access to Corvina Cloud are provided by Exor

Click on gear icon to insert credentials

- Possibilty to define different access profiles



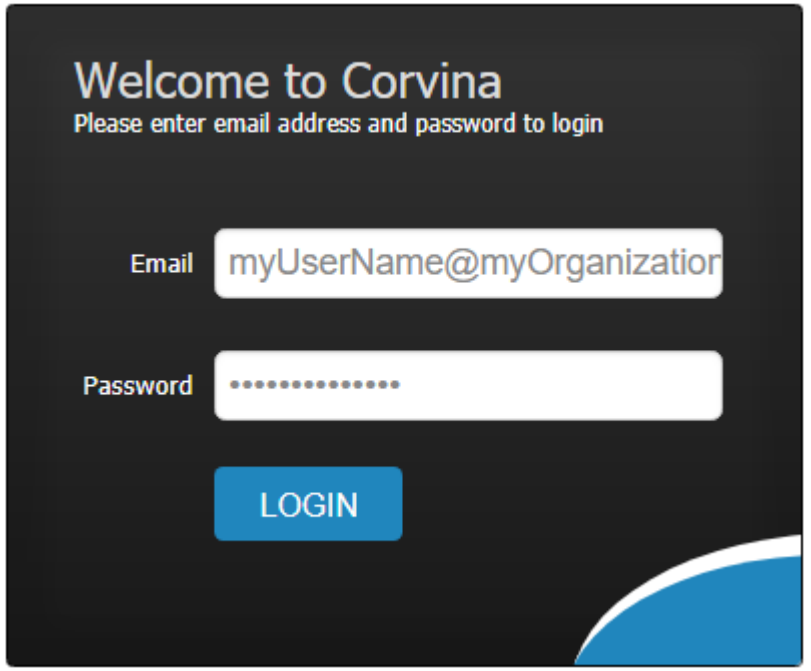
Access to Corvina Cloud

Username and password to access to Corvina Cloud are provided by Exor

Click on gear icon to insert credentials

- Possibilty to define different access profiles

Same credentials can be inserted in corvinacloud.com



Add a Device

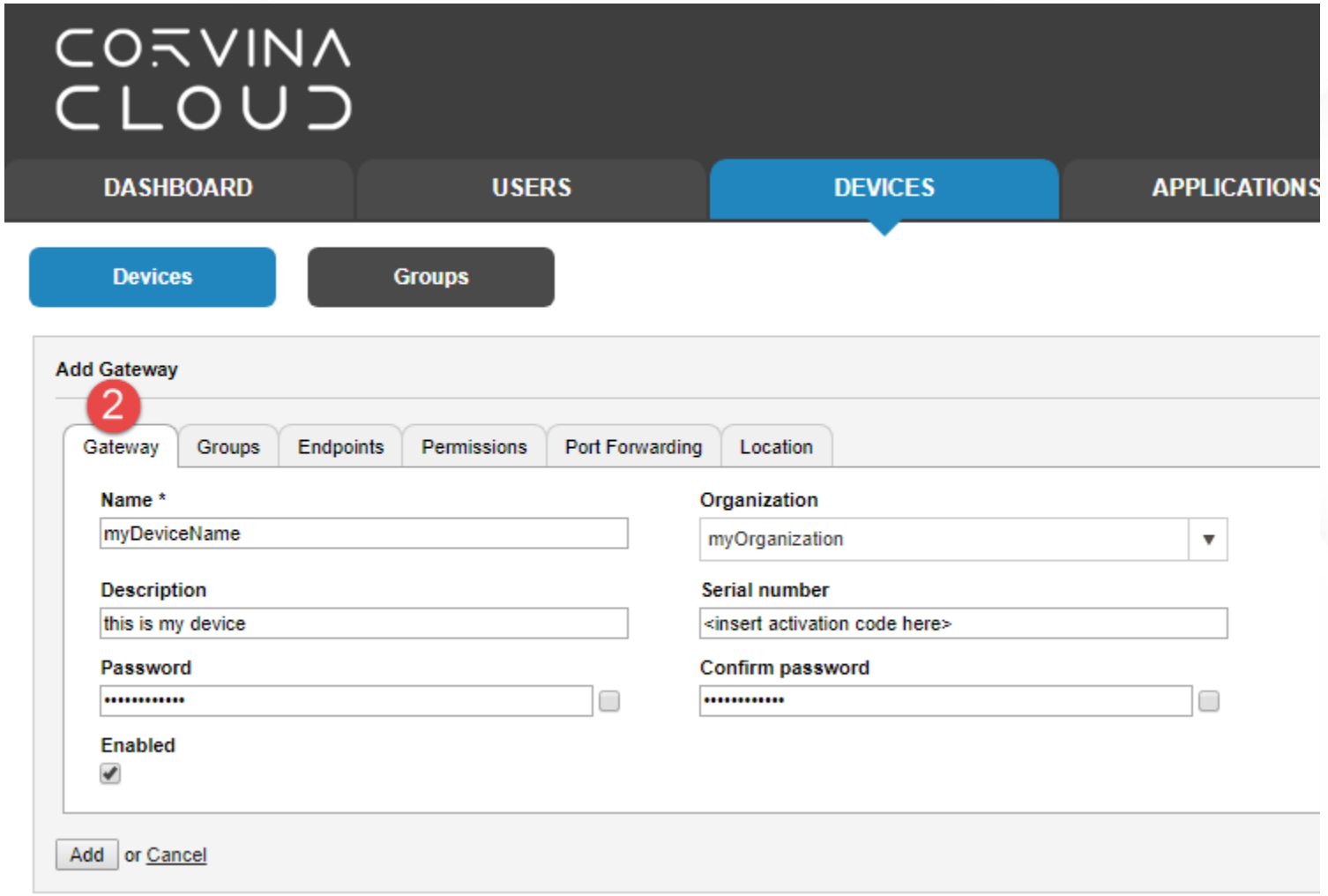
From Device tab

click on **Add gateway**

Add a Device

Fill details in **Gateway** tab:

- Name
- Organization
- Password
- Activation code

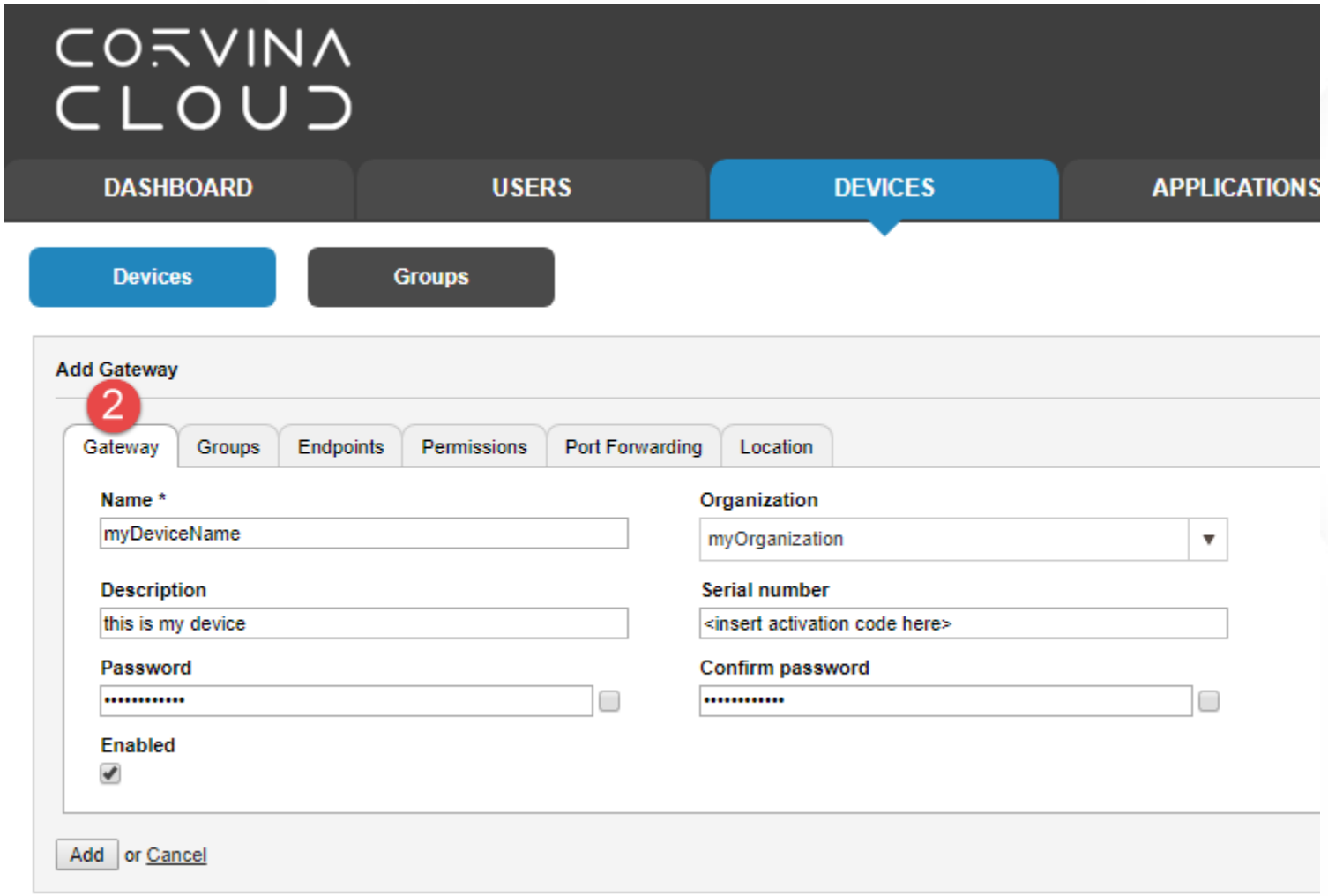


Add a Device

When configuring Corvina connection on the HMI use one use one of the following syntax to authenticate device

Name@Organization

Name/Organization

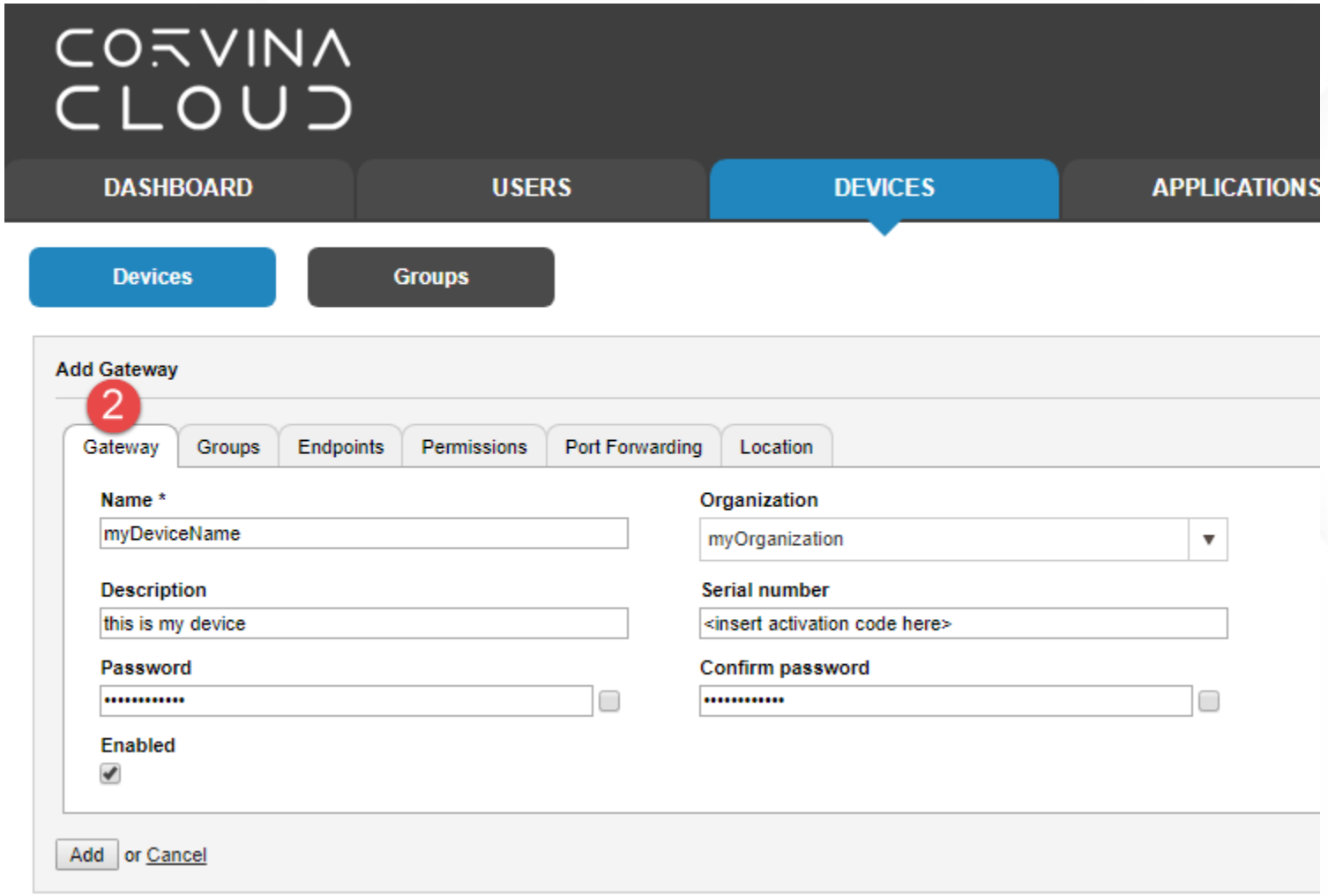


Add a Device

Serial Number field is used to certificate Device in Corvina Cloud server

For eX, eTOP, eSMART **HMI Cloud Enabler activation code** must be used

For eXware, PLCM07 **Serial Number** printed on HMI label must be used



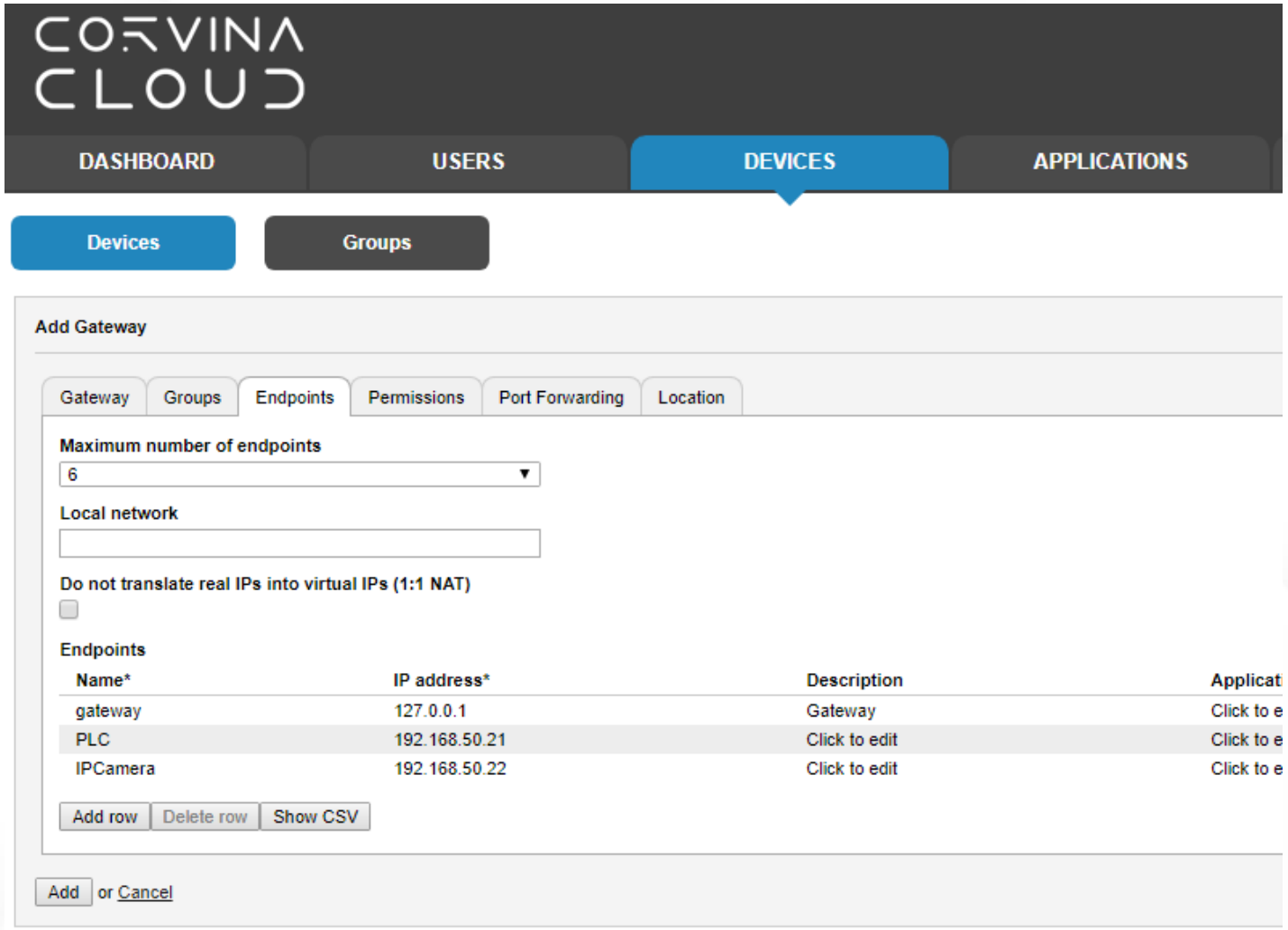
Add a Device

Configure **Endpoints**

Definition of any network point to be reachable from remote

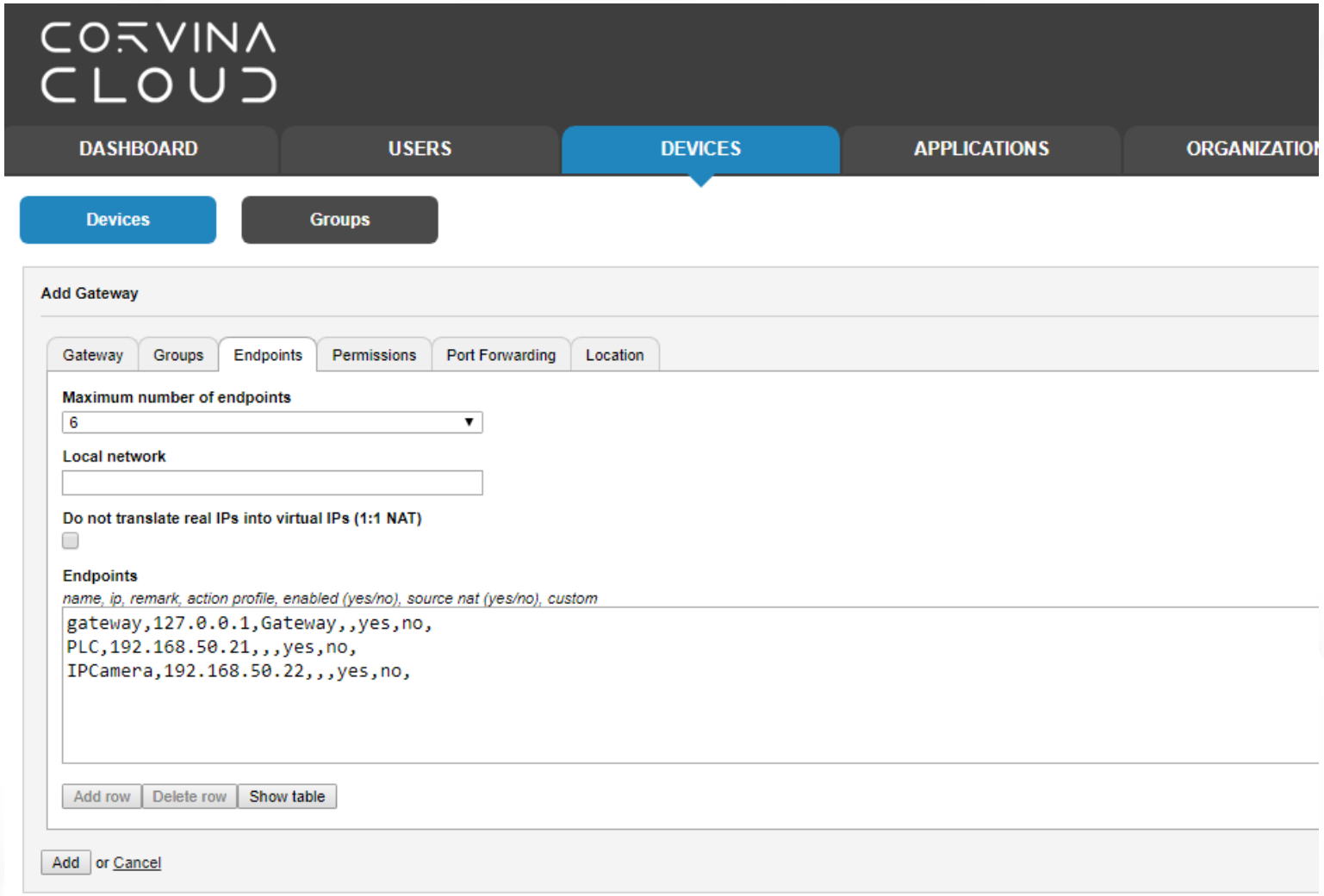
Possibility to open an entire network segment with **Local network** field using CIDR notation

Example: 192.168.100.0/24



Add a Device

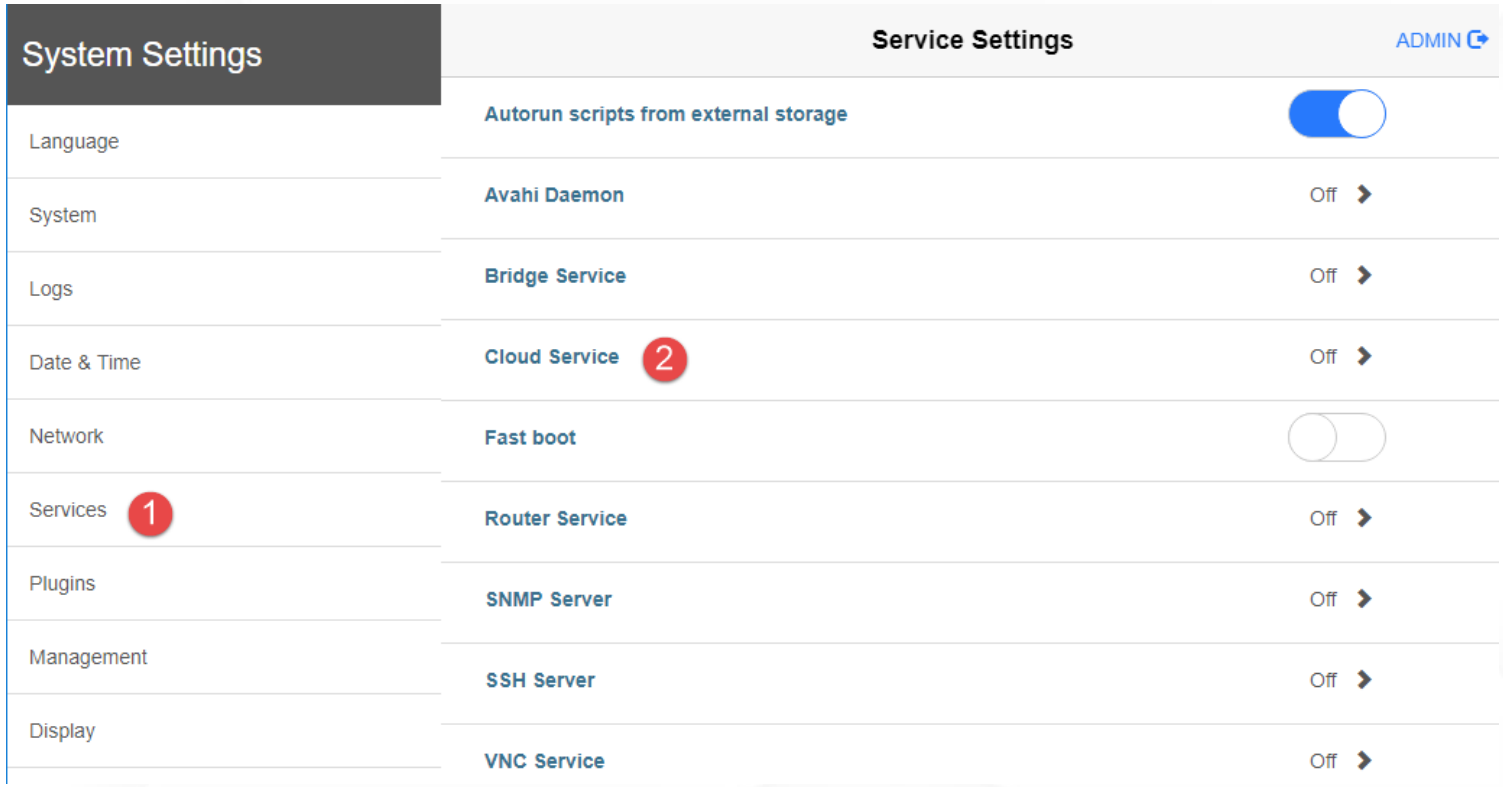
Possible to show endpoints as **CSV** for easy copy/paste between devices



Configure HMI service

From web browser go to

https://<IPaddress>/machine_config



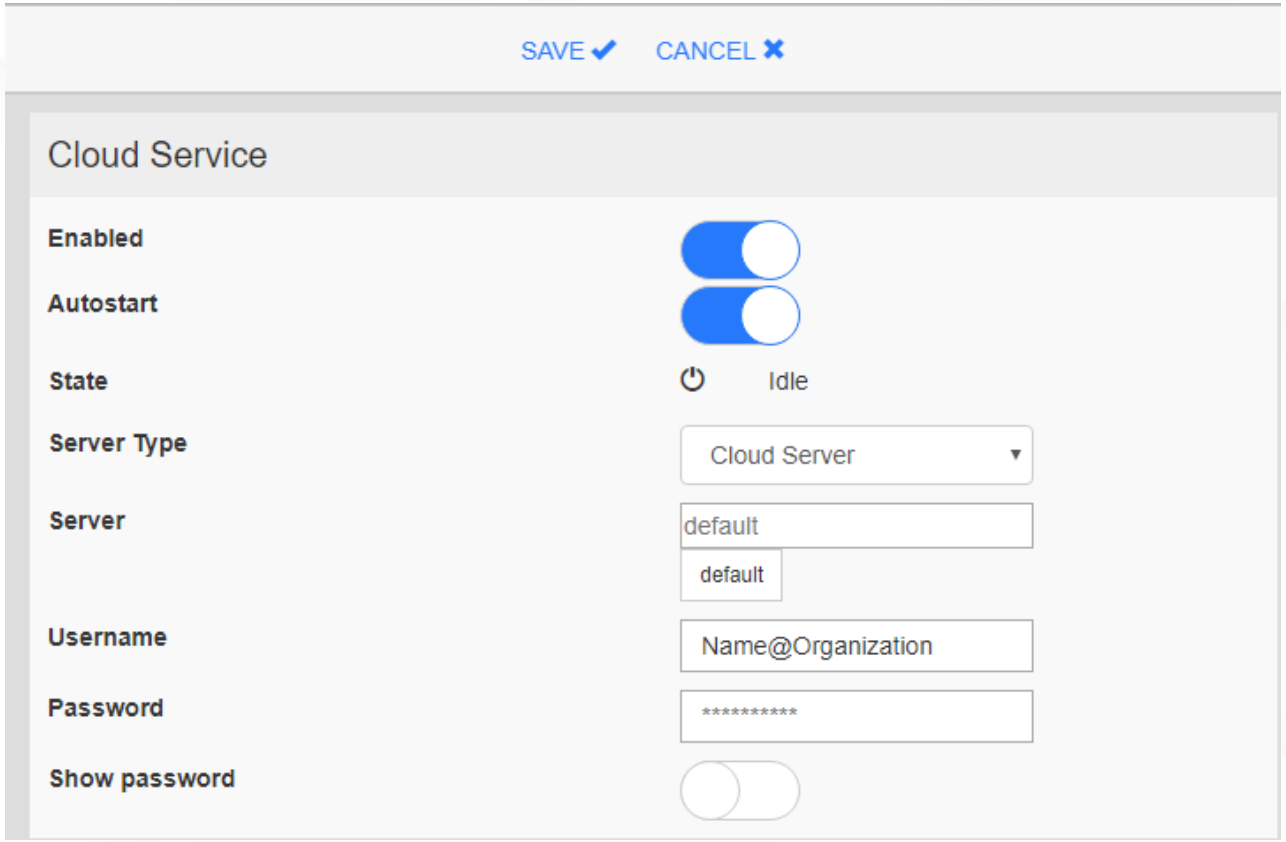
Configure HMI service

Enter in **Edit** mode

Switch to **Enabled**

Insert **Username*** and **Password**

Save



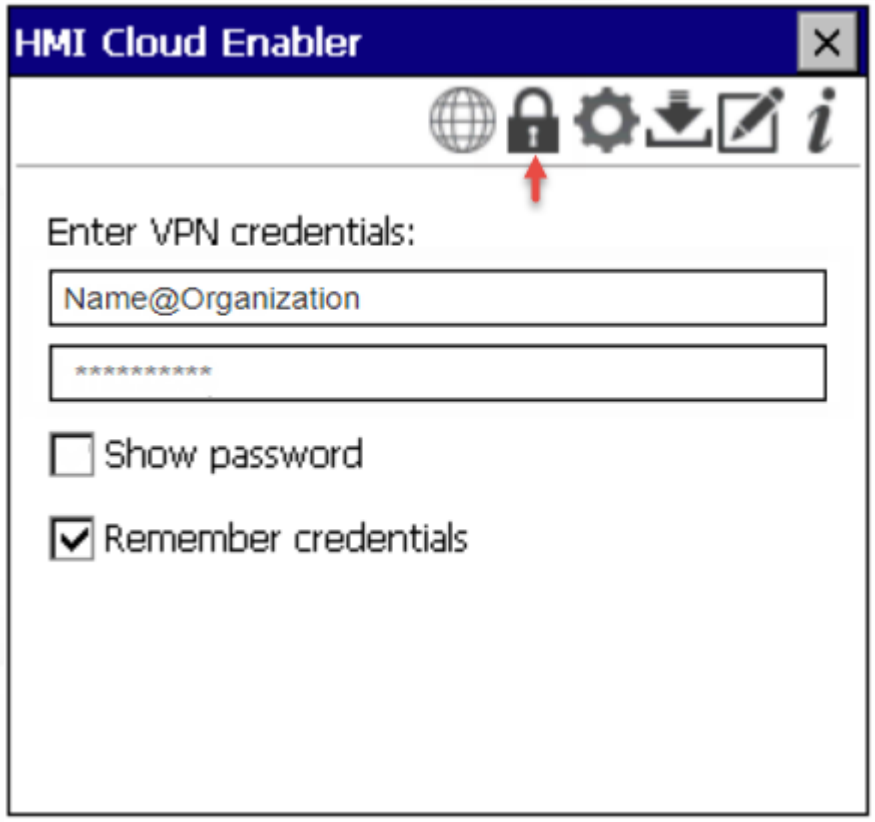
* Use one of the following syntax
  Name@Organization
  Name/Organization

Configure HMI service

In eTOP500/600 series (WCE-based)

Credenentials must be entered in HMI

from **HMI Cloud Enabler**

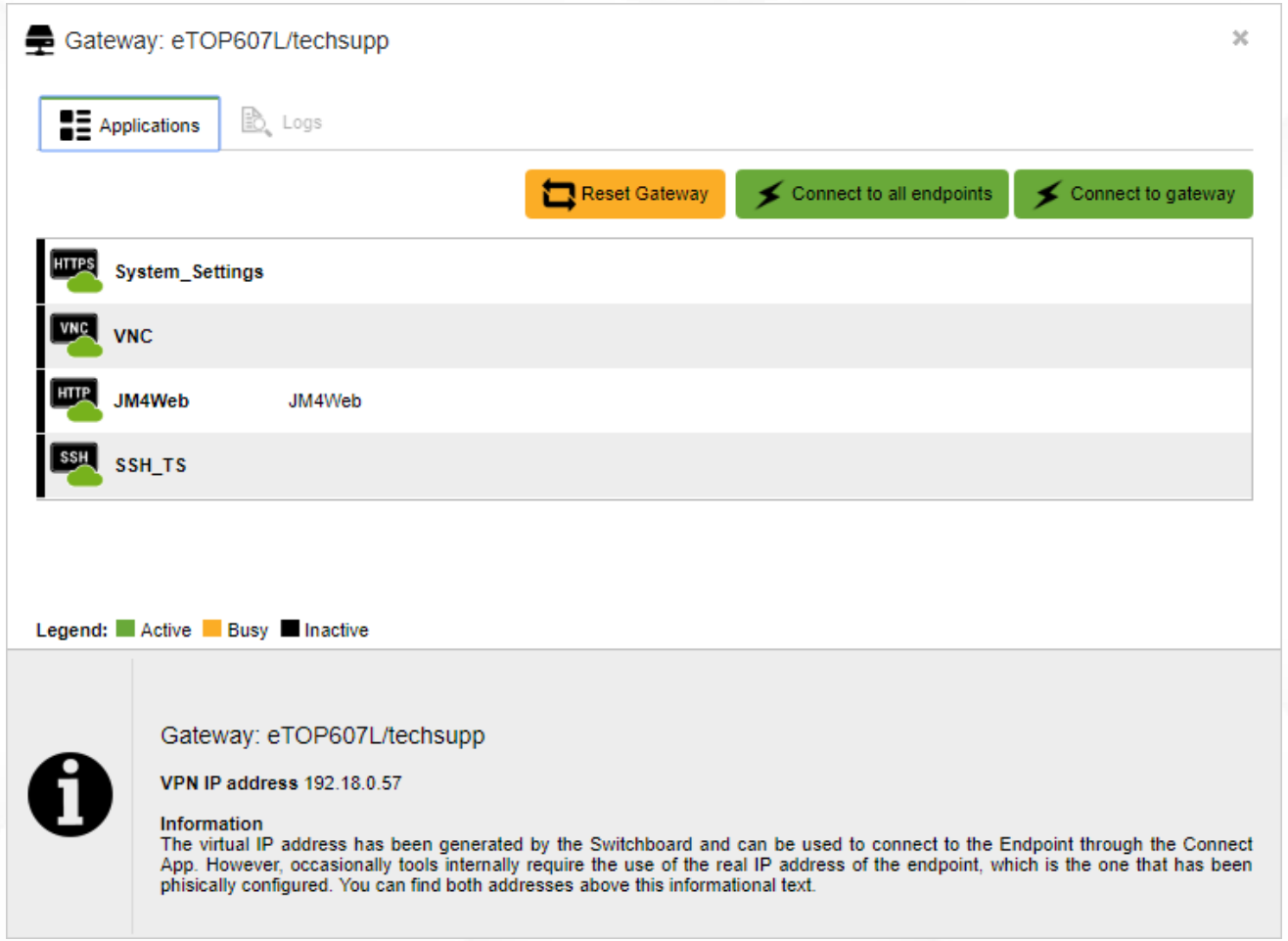


Configure Applications

Use **Applications** as a way to recall common usages in remote connections

Any external application

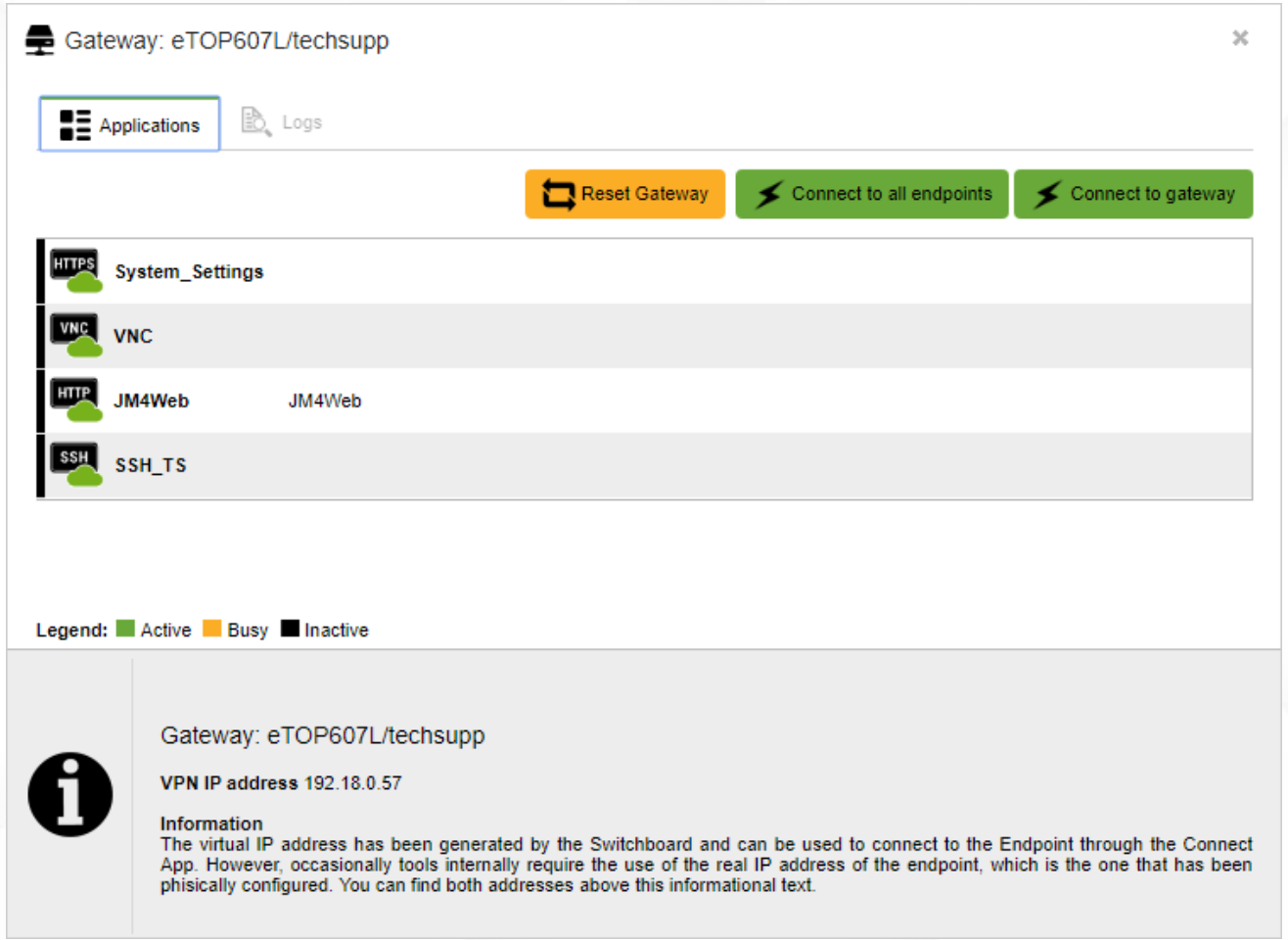
Embedded VNC and JM4Web (also from web access)



Configure Applications

Different type of Applications

SSH: open an SSH session
RDP: open a Remote Desktop Win32
VNC: open a standard VNC channel
Telnet: open a Telnet session
HTTP/S: open HTTP/S session
Custom: open external program

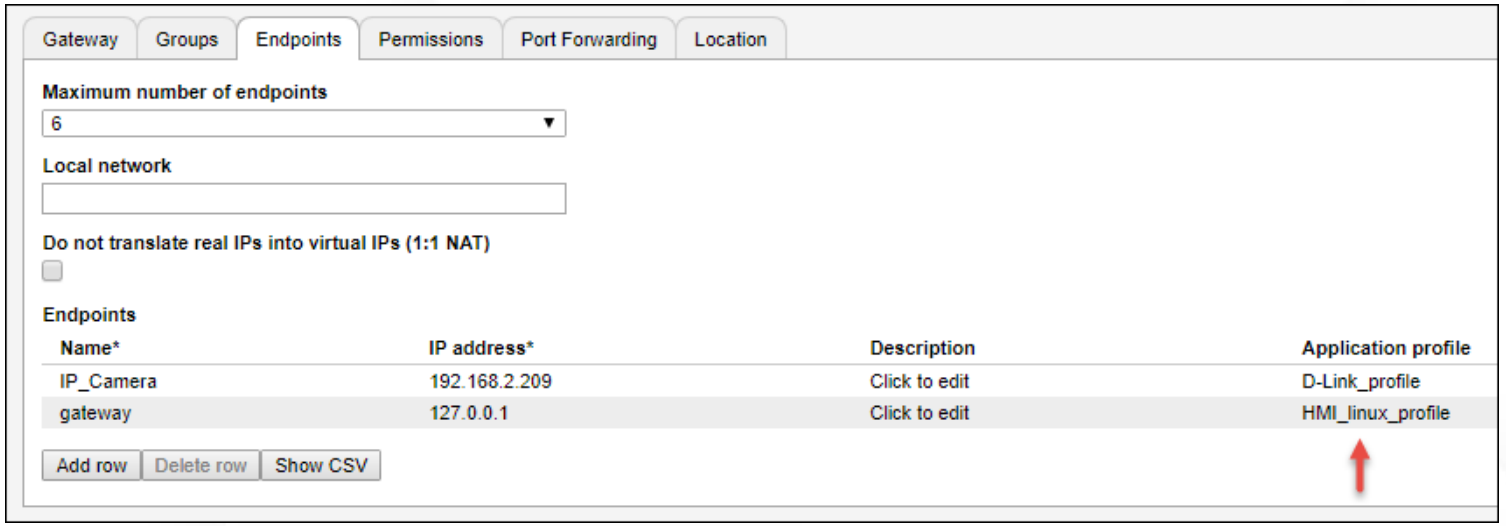


Assign Applications to Endpoints

To use Applications

Profiles must be created as group of Applications

Assigned to any Endpoint



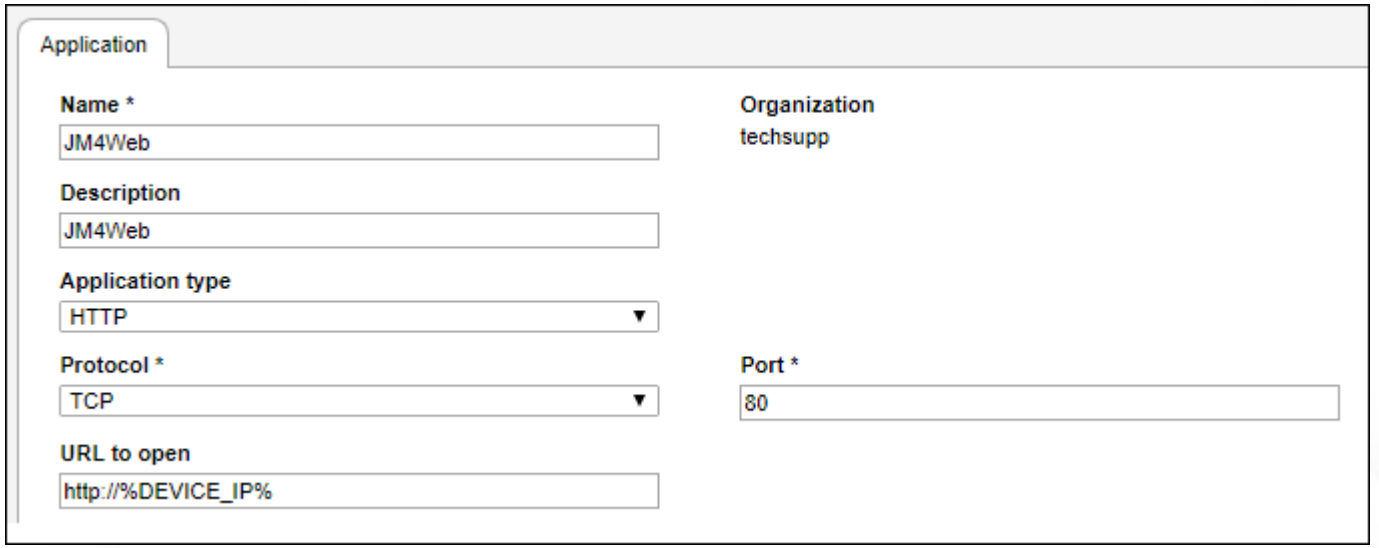
Use Cases – JM4Web

JM4Web Application
to point to JMobile web pages

Available from

Corvina Cloud App

Web portal
(smartphone, tablet
desktop browser)



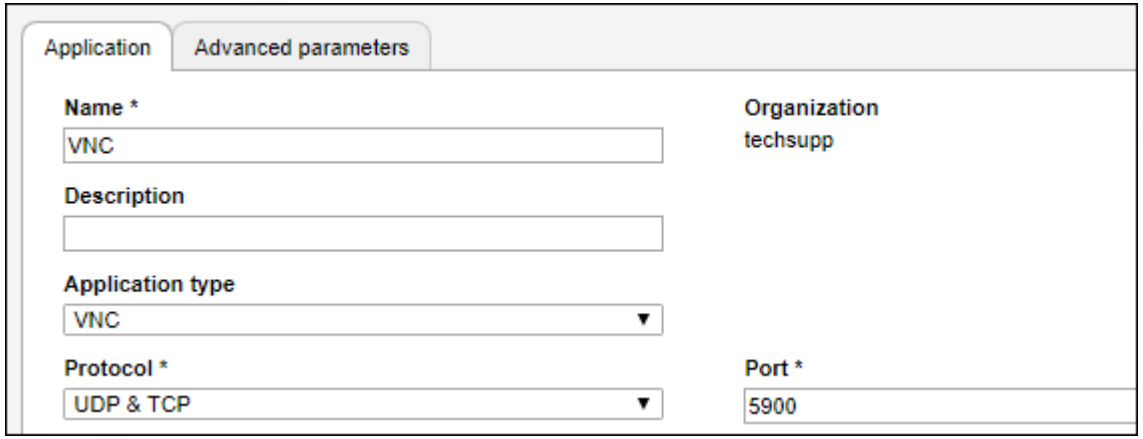
Use Cases – VNC

VNC Application
to point to HMI display

Available from

Corvina Cloud App

Web portal
(smartphone, tablet
desktop browser)



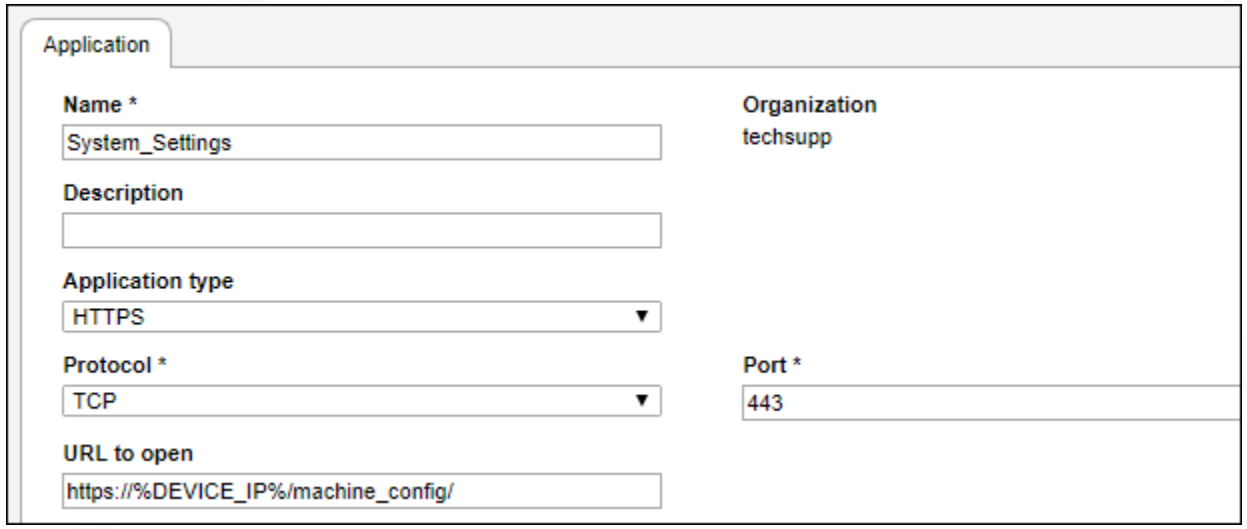
Use Cases – System Settings

System Settings Application
to point to HMI System Settings *

Available from

Corvina Cloud App

Web portal
(smartphone, tablet
desktop browser)

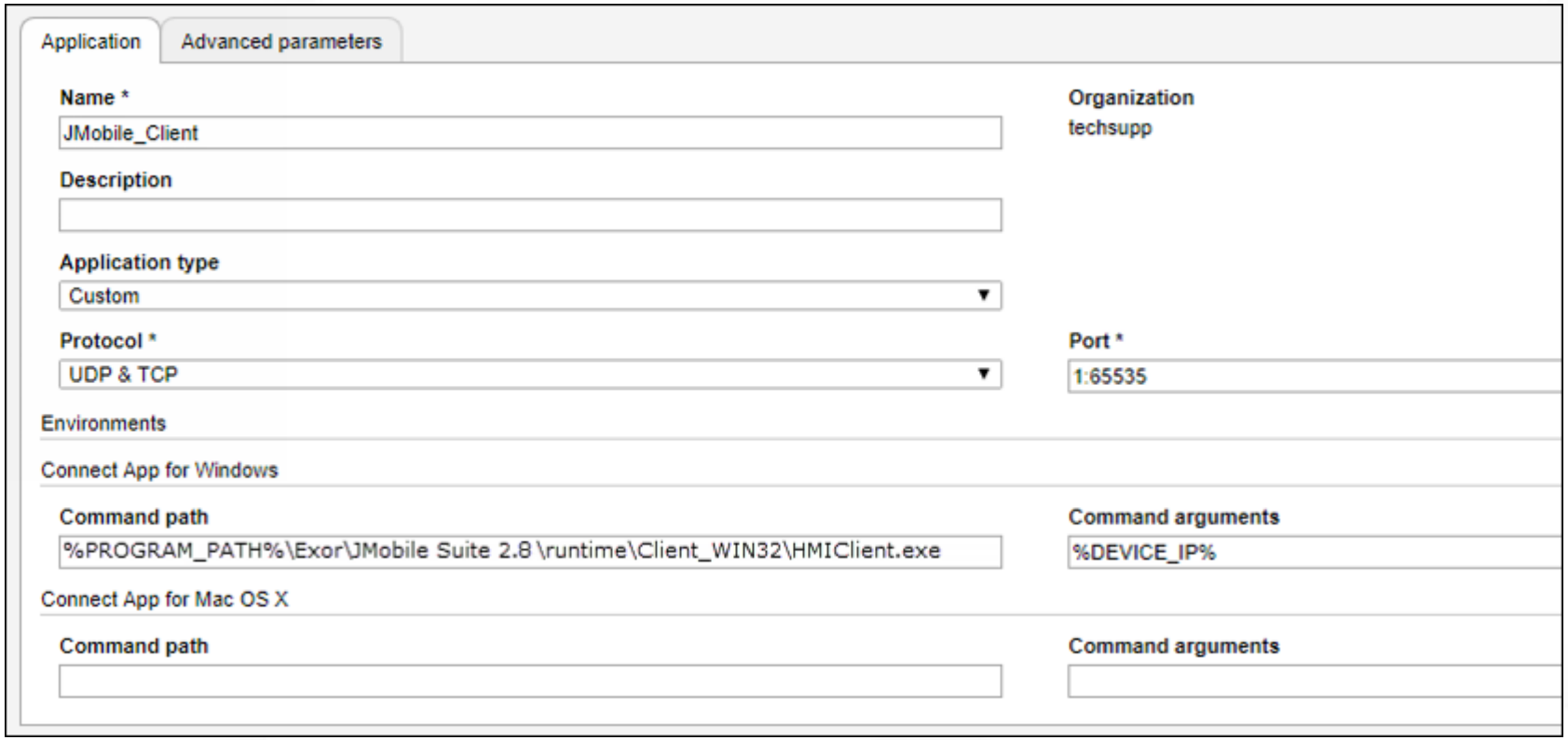


* For Linux-based HMI only

Use Cases – JMobile Client

JMobile Client Application to point to JMobile Runtime using JMobile Client

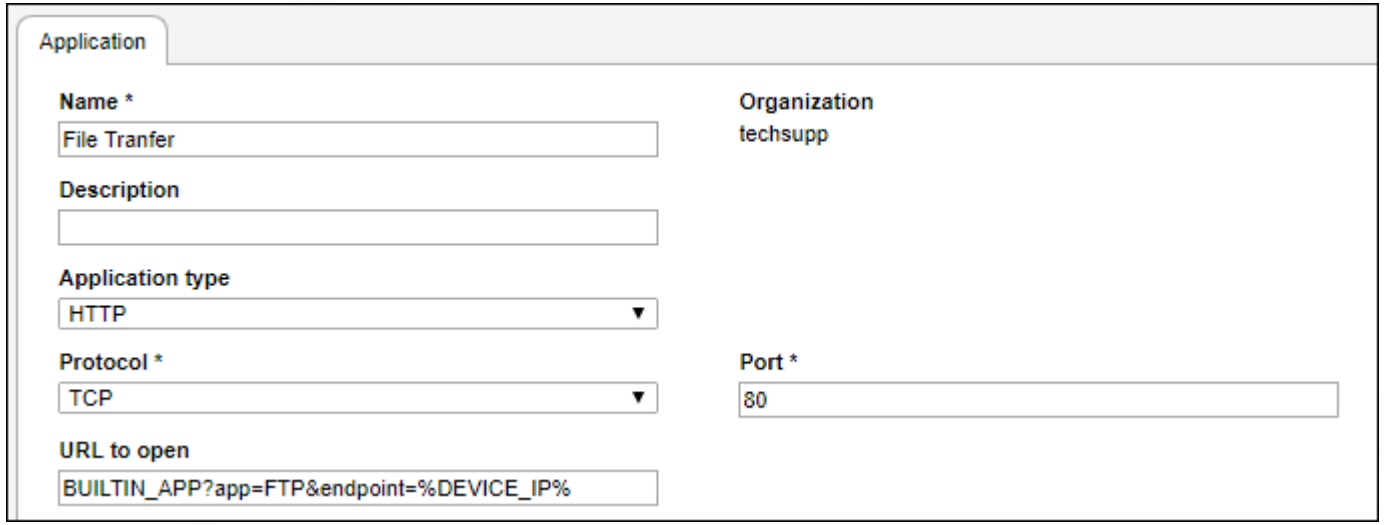
Launches exernal application



Use Cases – File Transfer

File Transfer Application
for easy remote/local file management

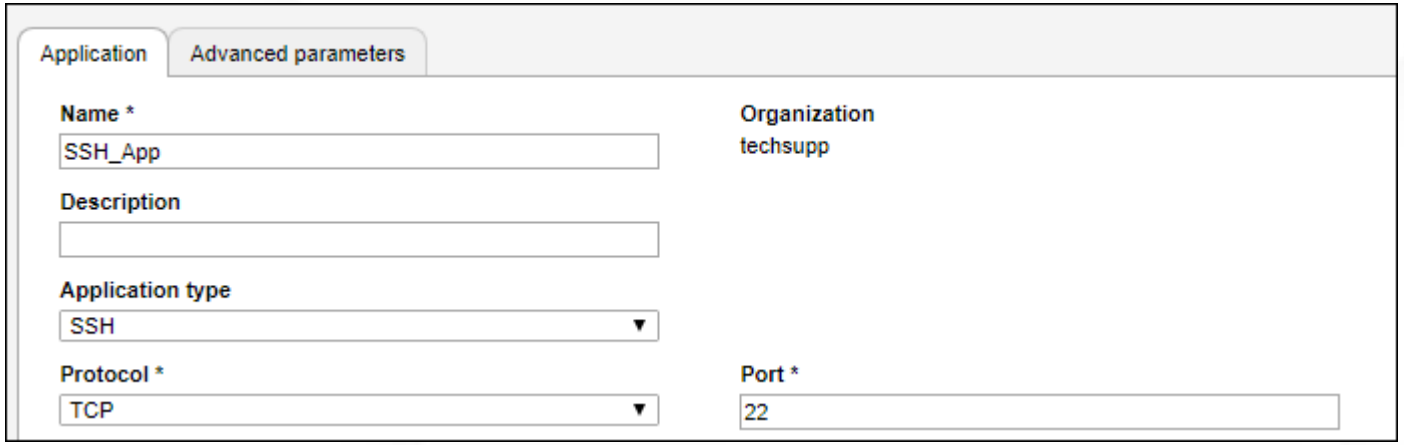
Uses a built-in application inside Corvina Cloud server



Use Cases – SSH

SSH Application
to open Linux shell *

Uses a built-in SSH Client



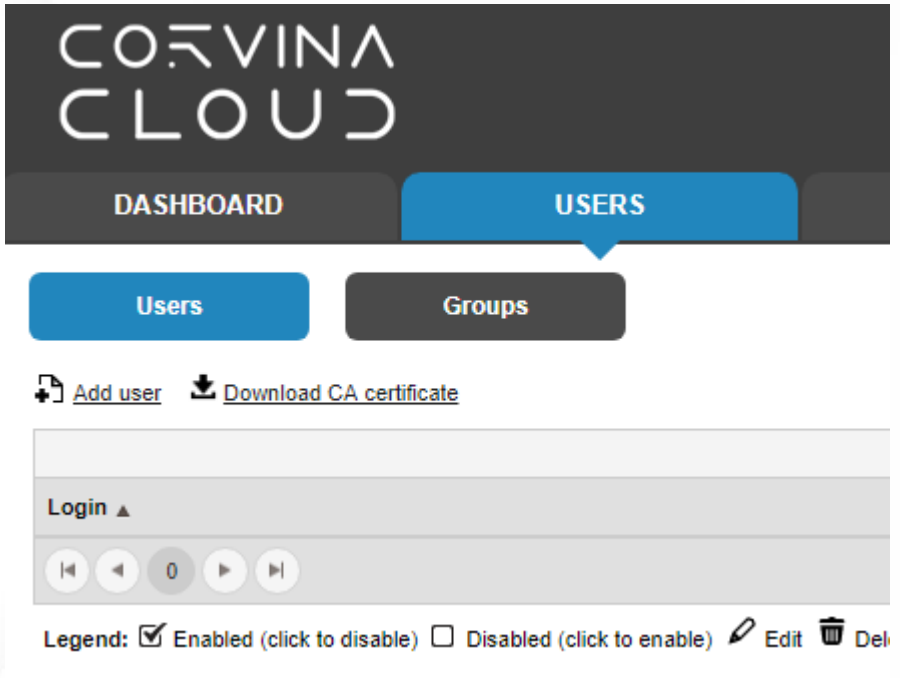
* Useful for debugging purpose. No need to install SSH Client on PC.

Add Users

Users can be added and can have different **Permissions**

Managers can add/remove items

Regular users can view items



Add Users

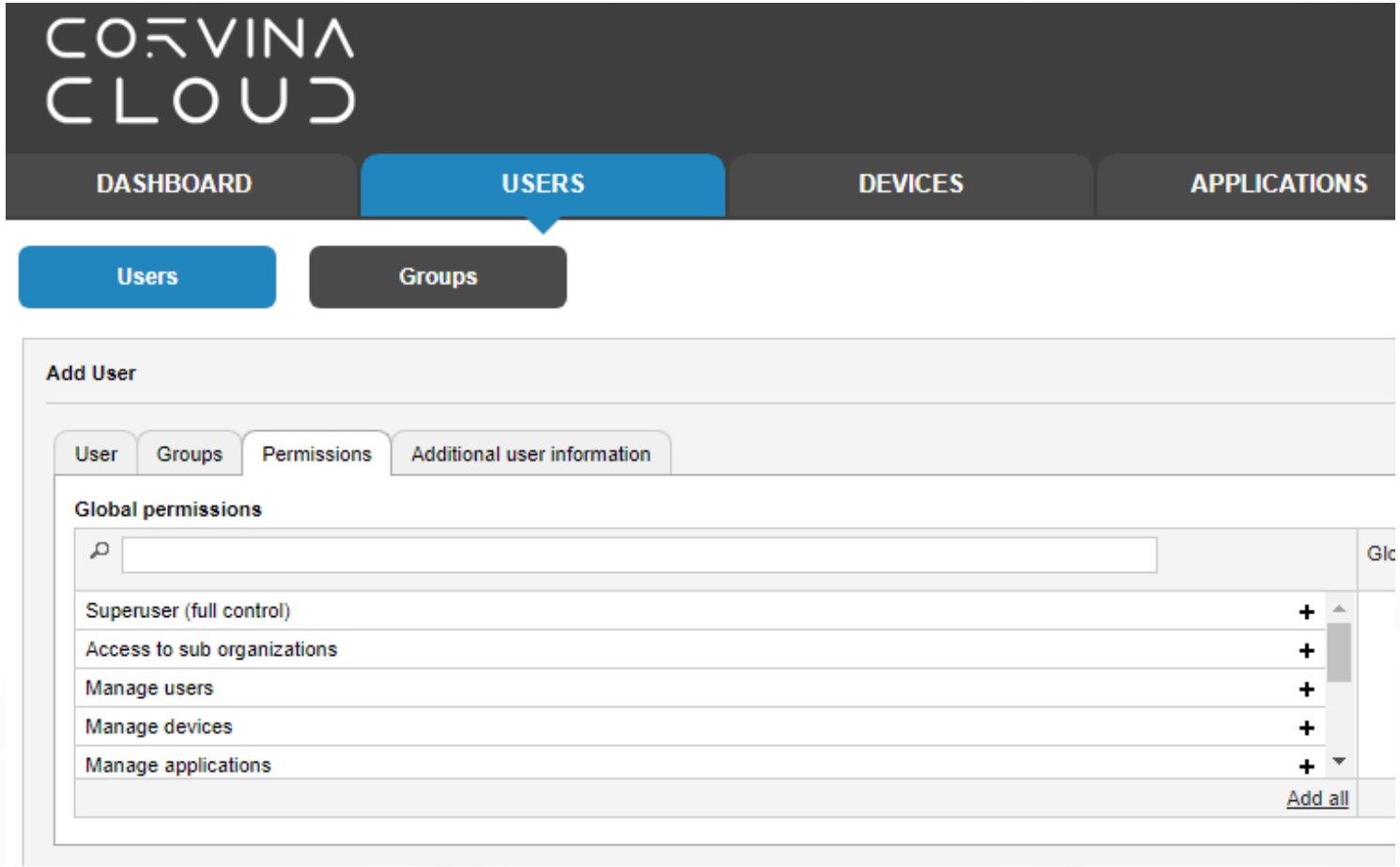
Fill details in **User** tab

- Name
- Organization
- Password

Add Users

Fill details in **Permissions** tab

Choose specific persmissions to give to user

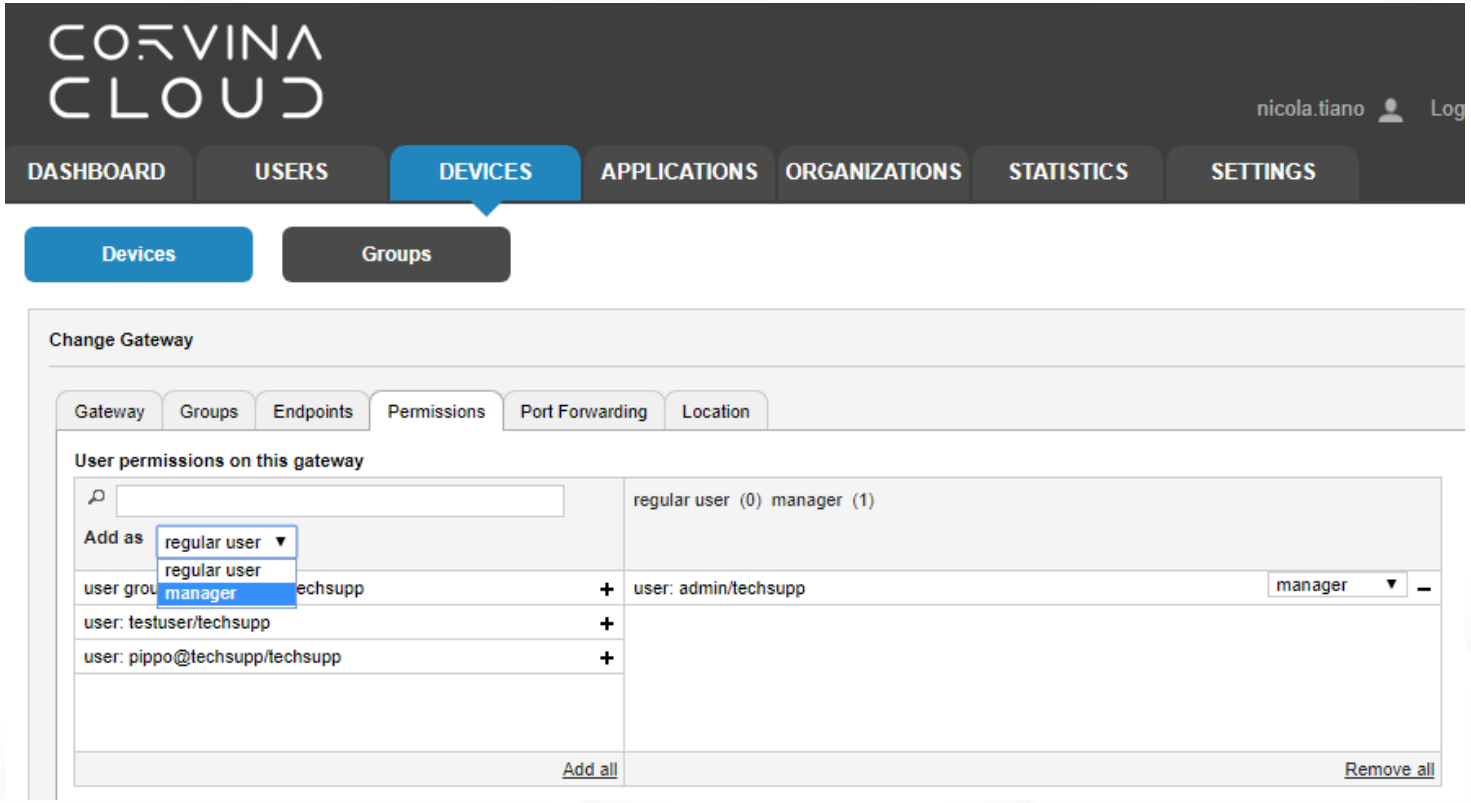


Specify permissions on devices

From device configuration
it is possible to specify user
permissions from
Permissions tab

Same concept can be applied
to user groups and
devices groups

Example:
Allow access to a group of device
to a group of users



JavaScript

JavaScript

A JavaScript function is executed when an event occurs.
For example, a user can define a script for the *OnMousePress* event.

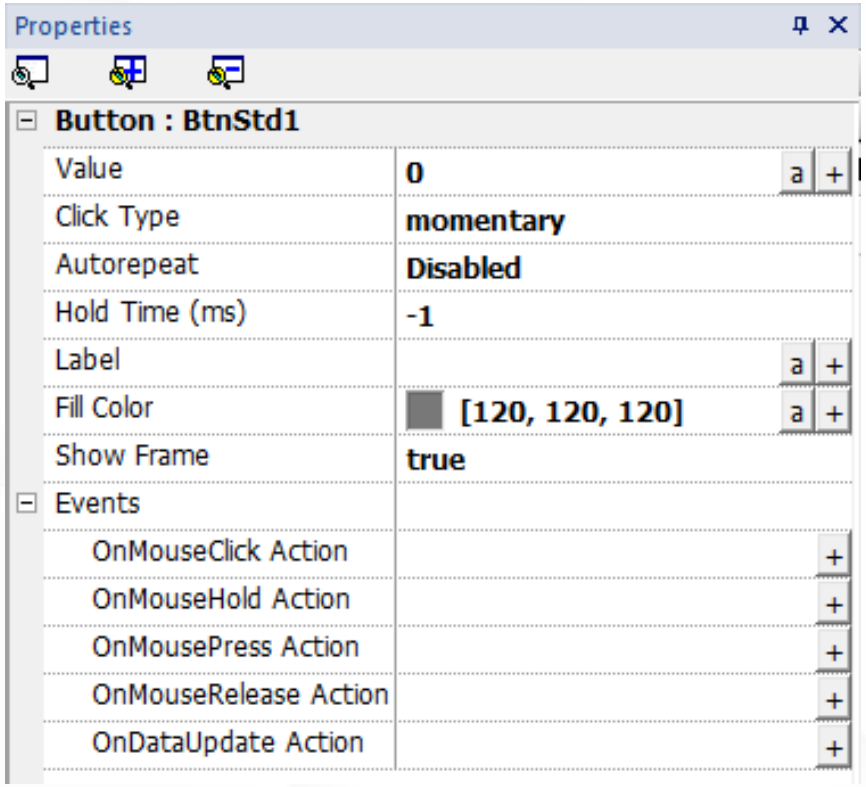
Evaluate carefully execution of JavaScript function at a given scan rate
with a scheduled action.
This approach minimizes the overhead required to execute logic on the HMI.

JavaScript engine run at the client side.
Each client connected to the server side of JMobile Runtime
(could be the client resident on HMI, or a JMobile Client)
will run the same script, providing output results that depend by client

Events

JavaScript Action called on Events

- Widget Events
  - OnMouse event
  - OnDataUpdate event
- Page Events
  - OnActivate
  - OnDeactivate
- System Events
  - Schedulers
  - Alarms



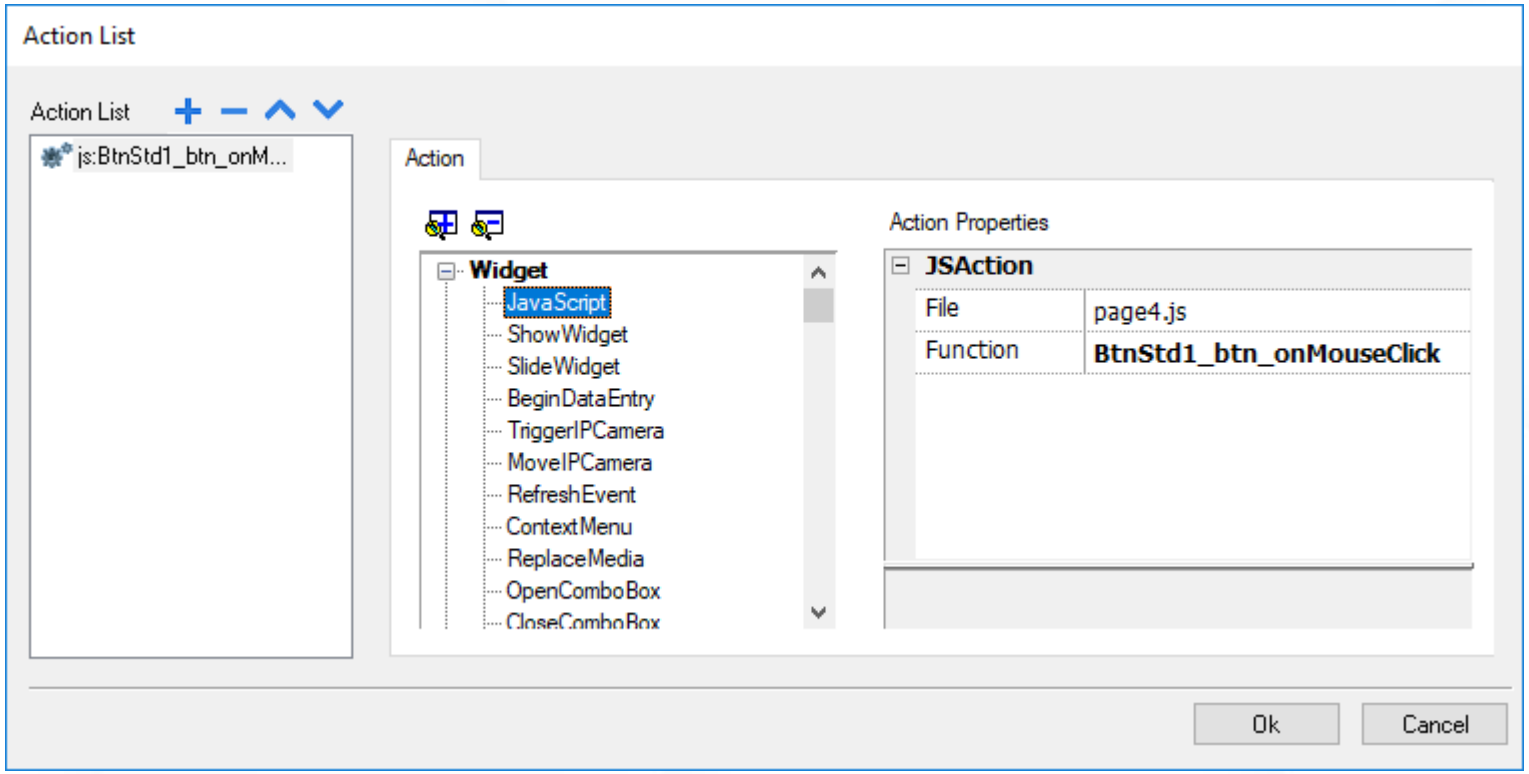
Events

JavaScript Action called on Events

Widget Events
- OnMouse event
- OnDataUpdate event

Page Events
- OnActivate
- OnDeactivate

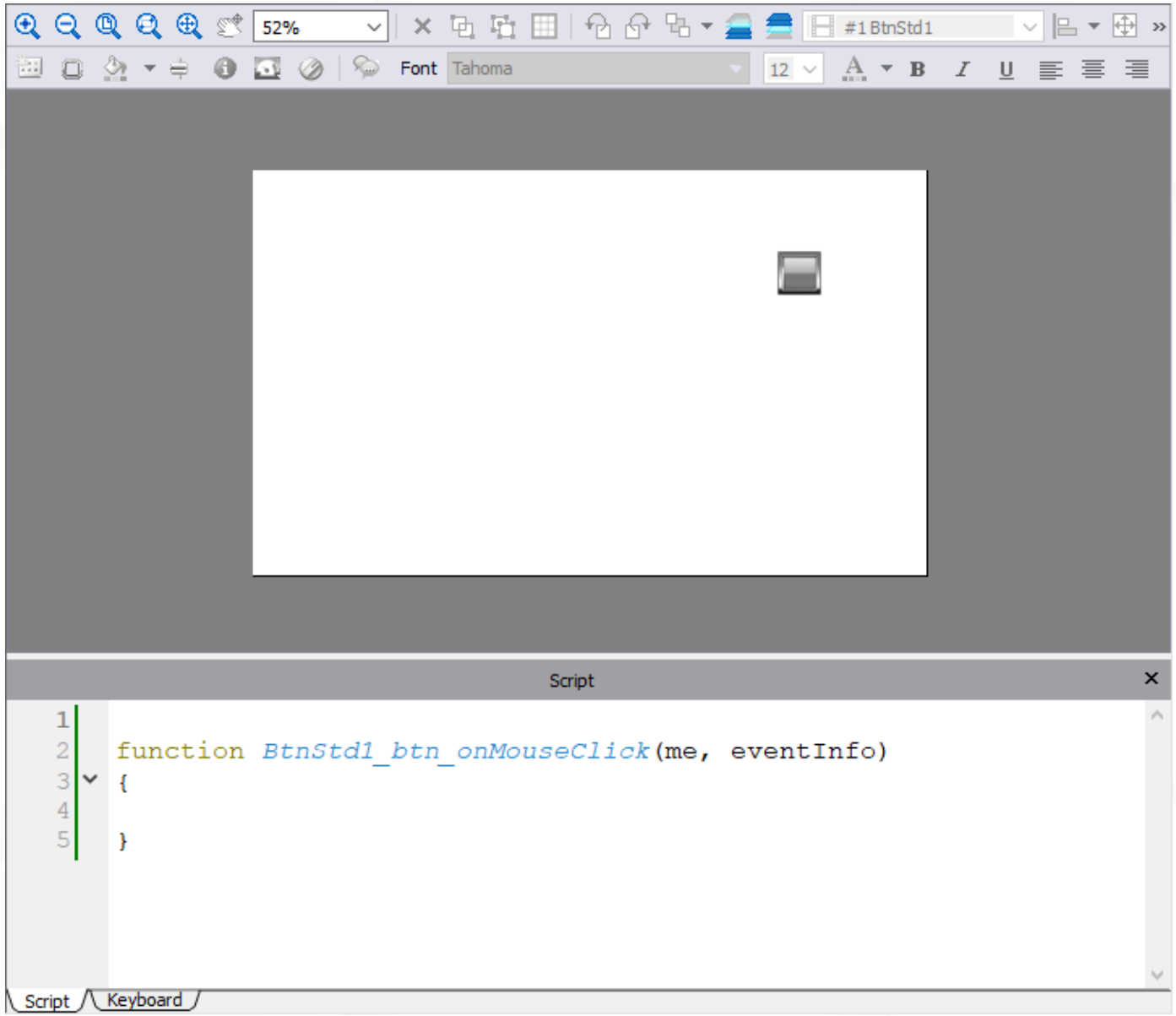
System Events
- Schedulers
- Alarms



Events

JavaScript Action called on Events

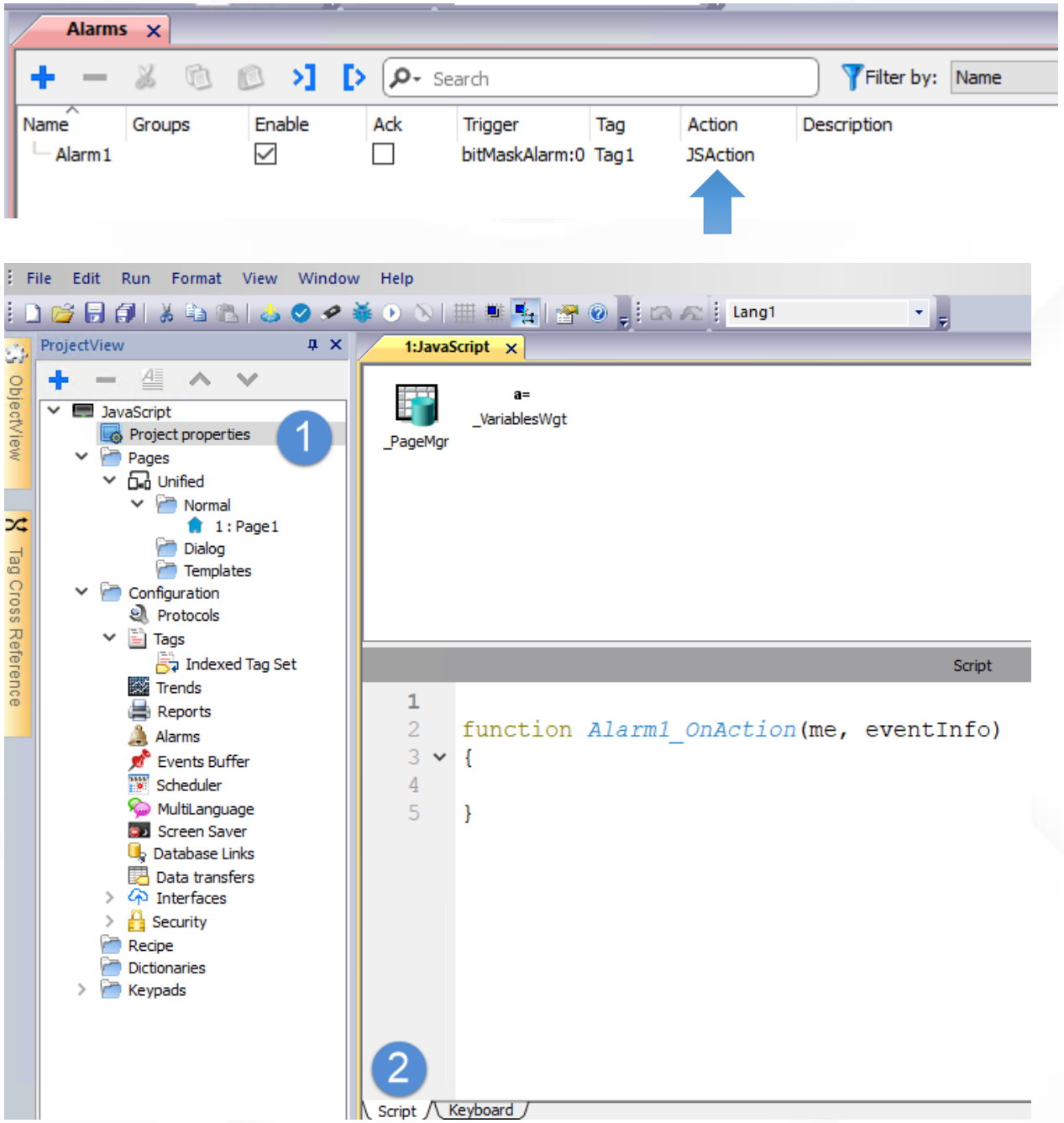
- Widget Events
  - OnMouse event
  - OnDataUpdate event
- Page Events
  - OnActivate
  - OnDeactivate
- System Events
  - Schedulers
  - Alarms



Events

JavaScript Action called on Events

- Widget Events
  - OnMouse event
  - OnDataUpdate event
- Page Events
  - OnActivate
  - OnDeactivate
- System Events
  - Schedulers
  - Alarms



Project Object

Page change and open dialog

```
project.nextPage();
project.prevPage();
project.homePage();
project.loadPage("Page5.jmx");
project.showDialog("Dialog.jmx");
project.closeDialog();
project.showMessage("Hi This is test message");
```

Project Object

Read and write Tags

```
var tagvalue = project.getTag("Tag1");

project.setTag("Tag1",123);
```

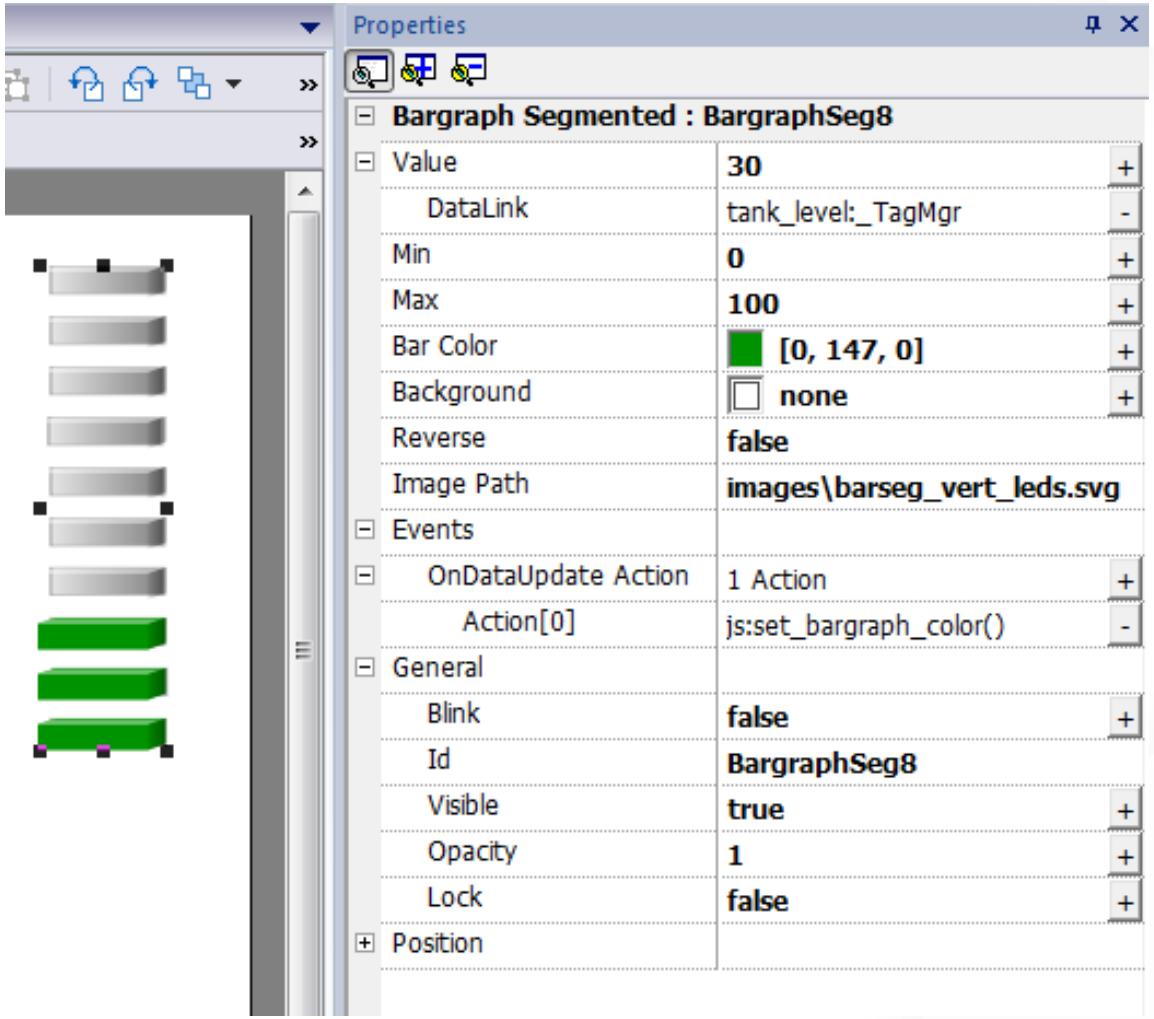
Page Object

Access and chance graphic properties

```
var page_height = page.getProperty("height");
page.setProperty("color", "rgb(255,0,0)");

var shape = page.getWidget("rect1");
var W = shape.getProperty("width");

var shape = page.getWidget("rect1");
shape.setProperty("width", 350);
```

NOTE: If properties are set via JavaScript, Static Optimization property must be set to Dynamic to allow dynamic rendering of static objects

JavaScript example explained

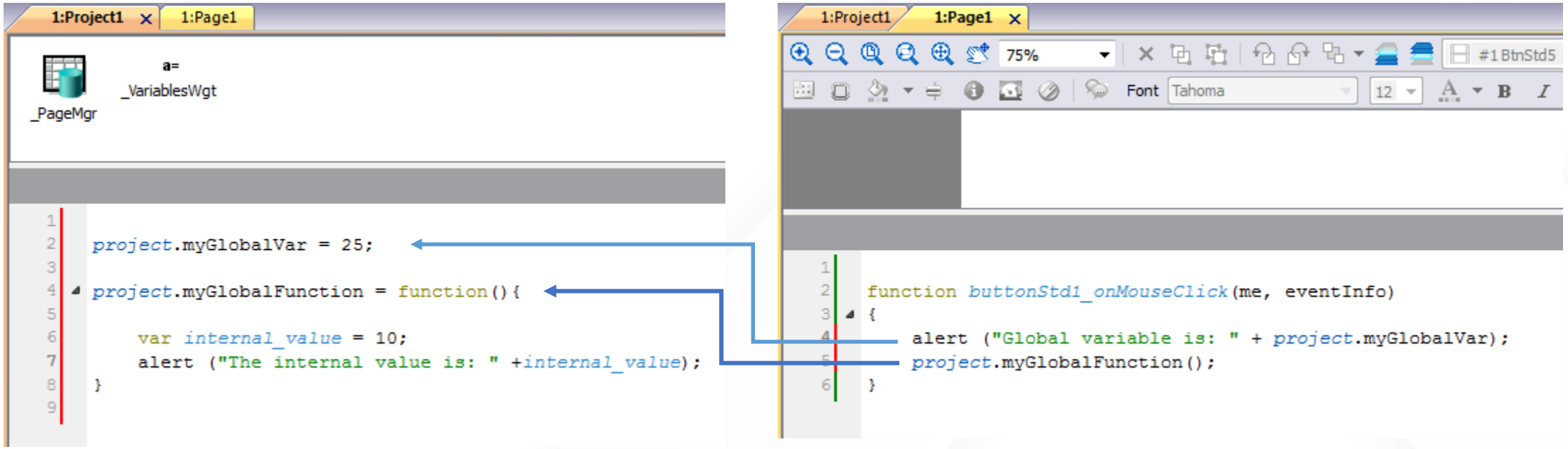



```
function set_bargraph_color(me, eventInfo){

    var tankstate = new State();
    var tankvalue = project.getTag("tank_level", tankstate, 0);
    var bargraph = page.getWidget("BargraphSeg8");

    if (tankstate.isQualityGood()){
            if (tankvalue>70){
                bargraph.setProperty("fill","rgb(255,0,0)");
            }
            else{
                bargraph.setProperty("fill","rgb(0,147,0)");
            }
    }
    else{
        bargraph.setProperty("fill","rgb(127,127,127)");
    }

    return false;
}
```

JavaScript shared code

"project" global variable can be used to share JavaScript code between pages

Variables and functions created into Project Properties tab, can be recalled from other project pages



JavaScript read/write files

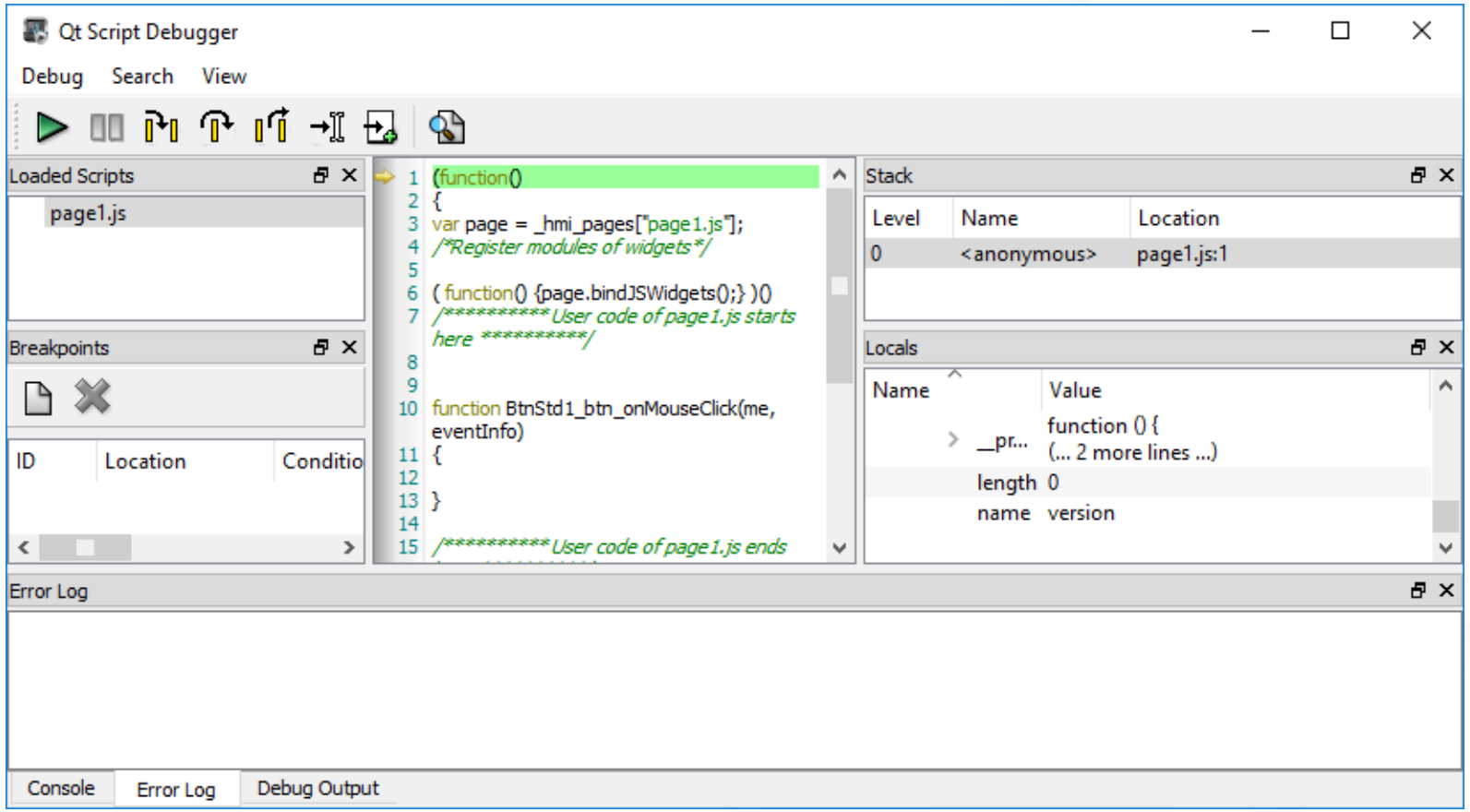
File management to read and write text files

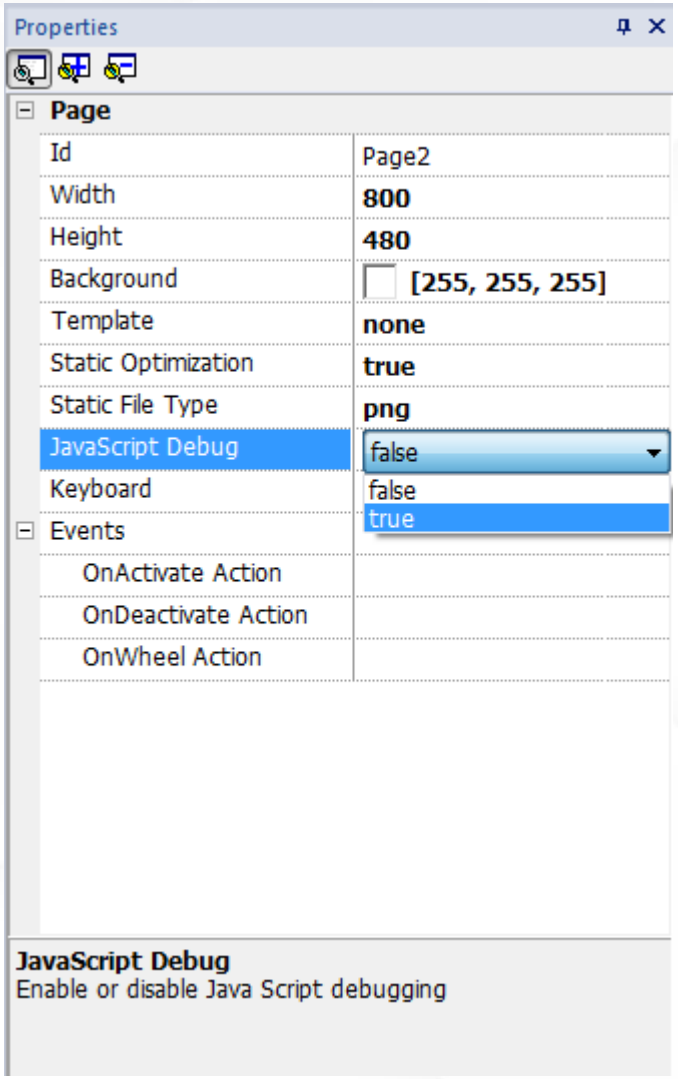
- Custom reports in text files or file management
- Parsing of existing text files and write on Tags

```
fs.writeFile(my_file,  project.getTag("text")+","+project.getTag("counter")+"\r\n", "r");

var file_read = fs.readFile(my_file);
```

Javascript Debug

QT Script debugger

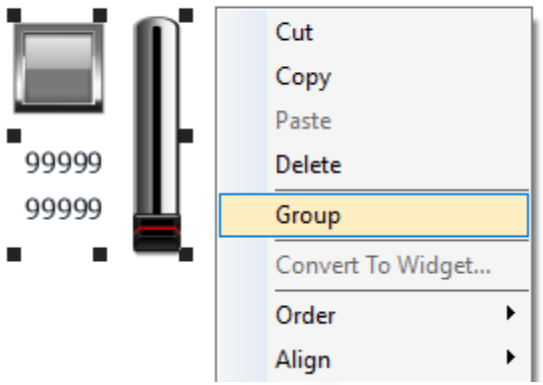


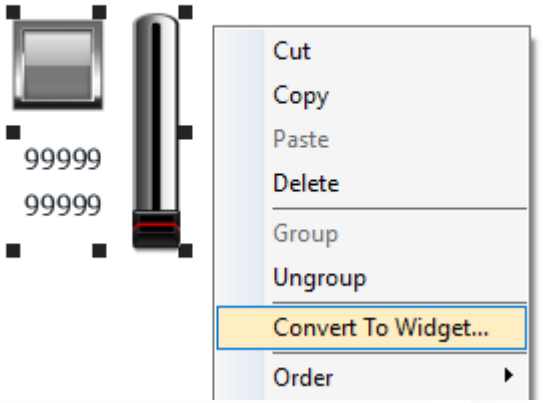


Custom Widgets and User Gallery

Custom Widgets

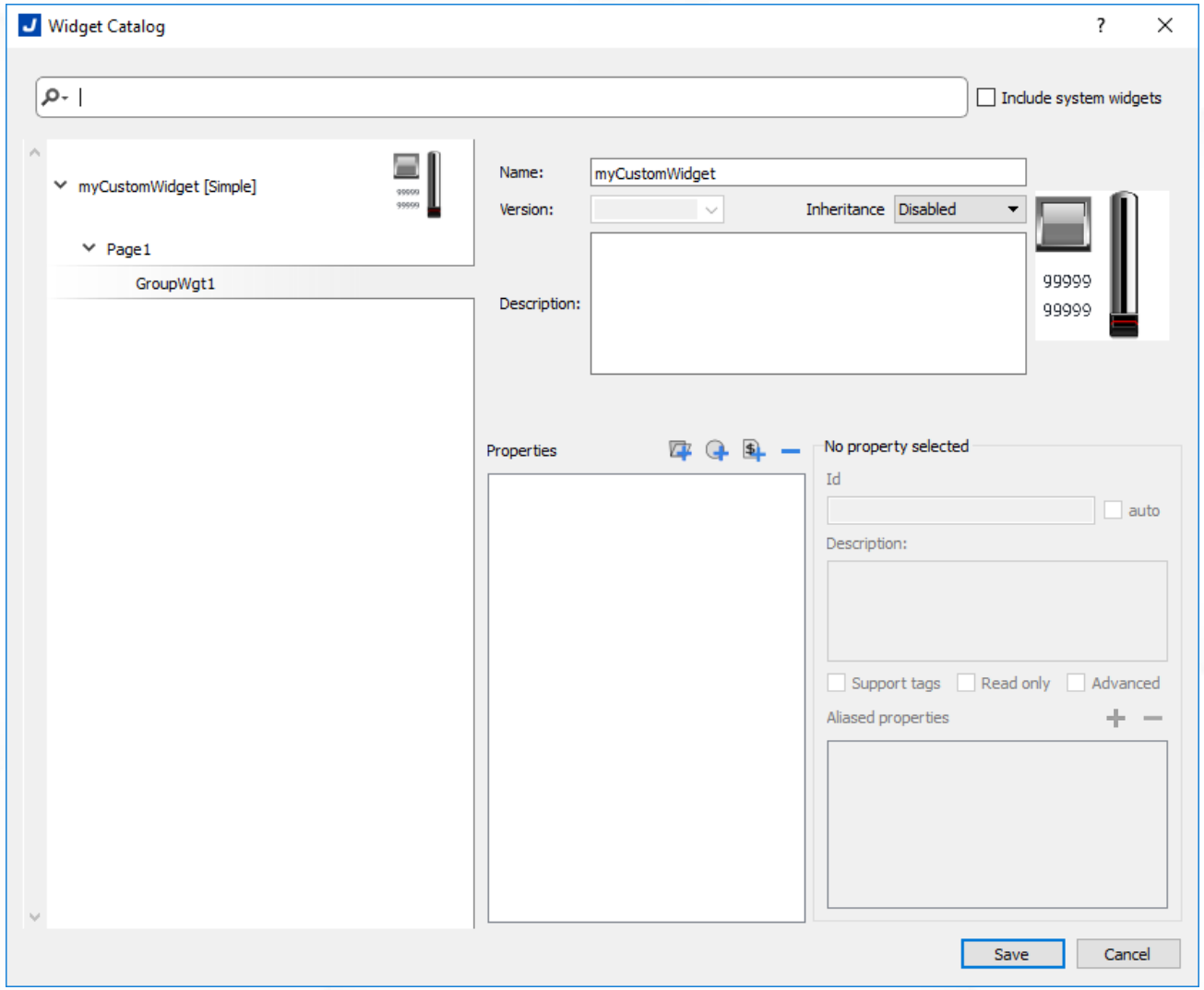
Group many Widgets,
then convert to Custom Widget





Custom Widgets

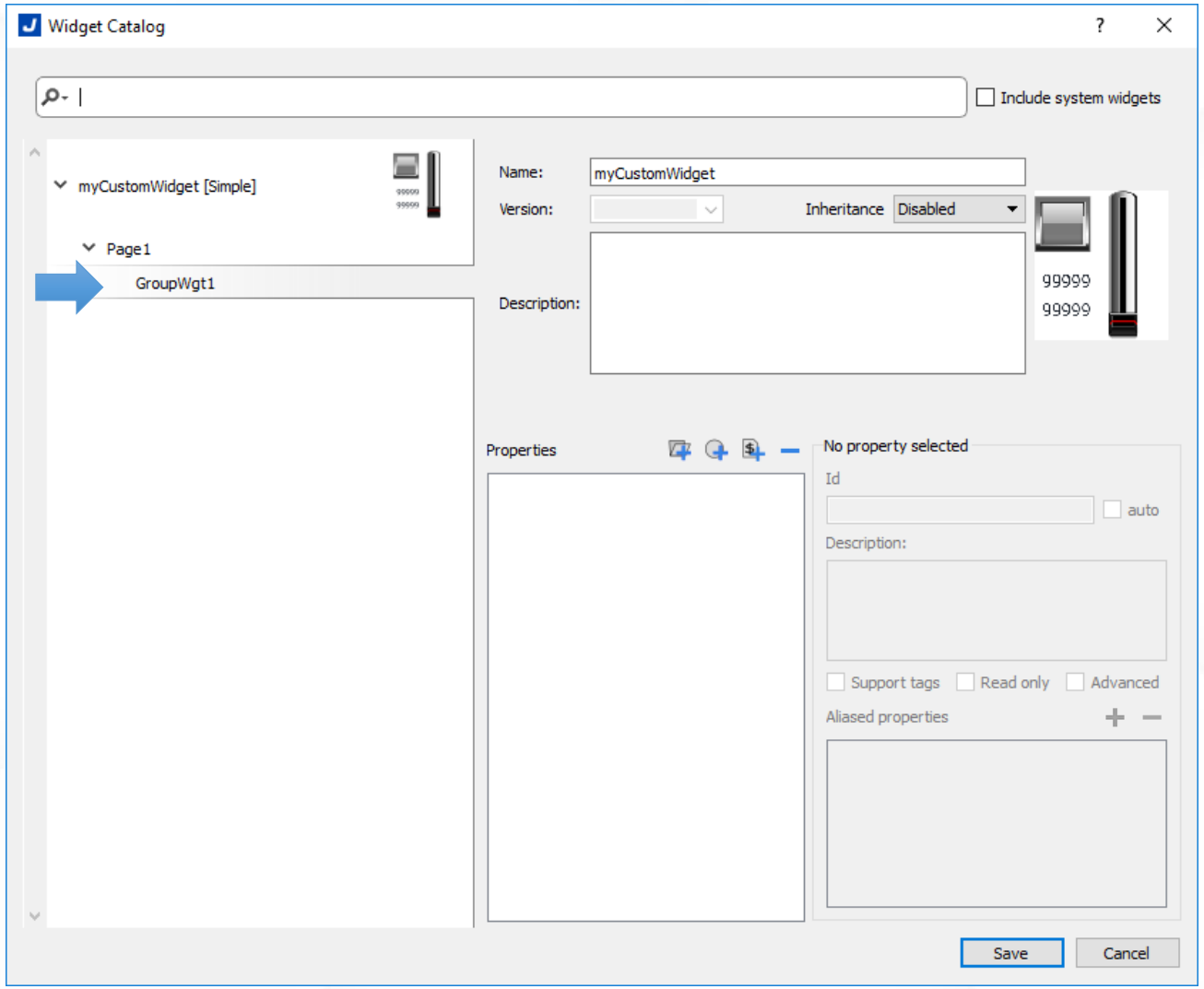
Create your own Widget Catalog



Custom Widgets

Create your own Widget Catalog

View where Custom Widgets are used in the project

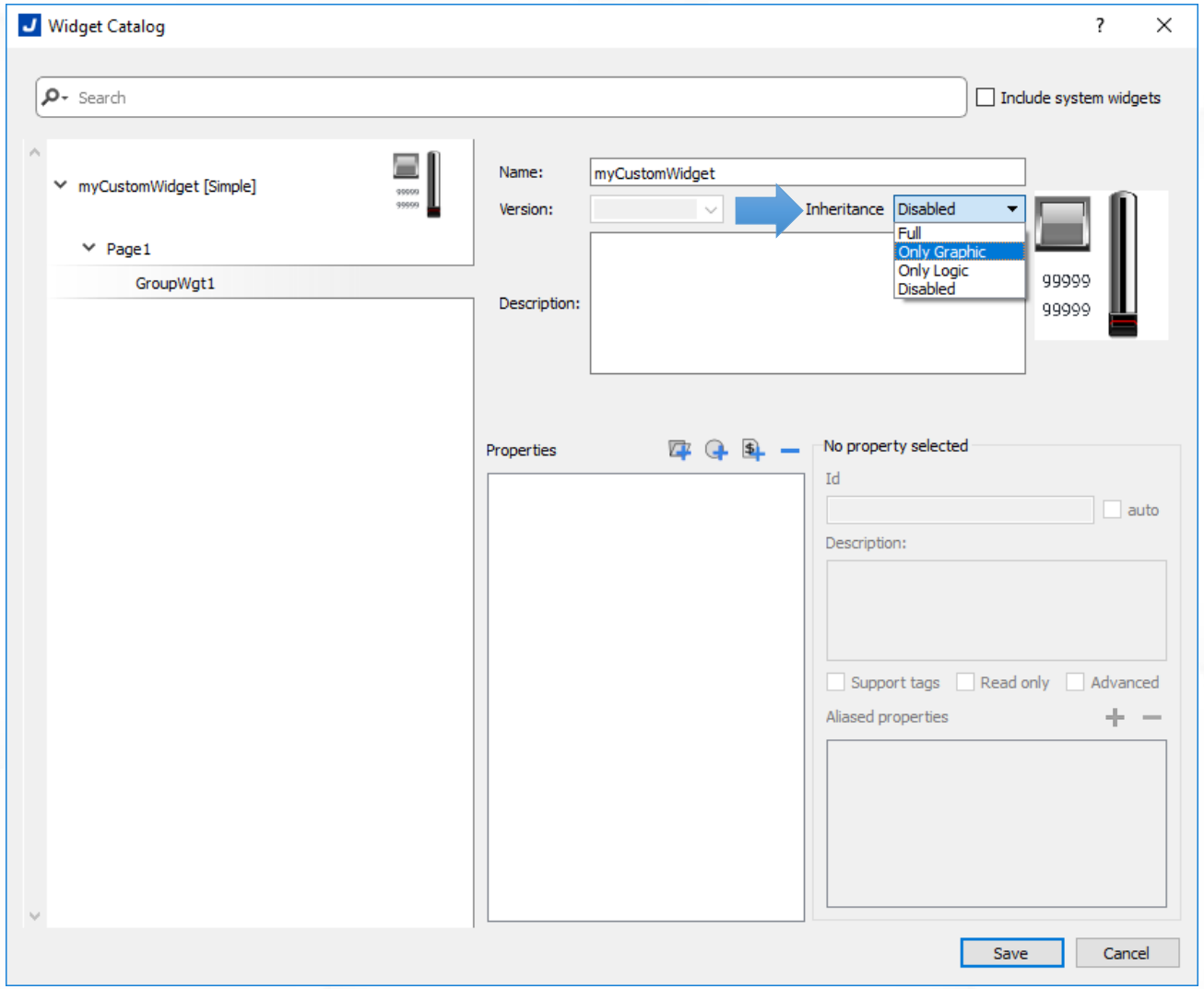


Custom Widgets

Create your own Widget Catalog

View where Custom Widgets are used in the project

Set inheritance configuration



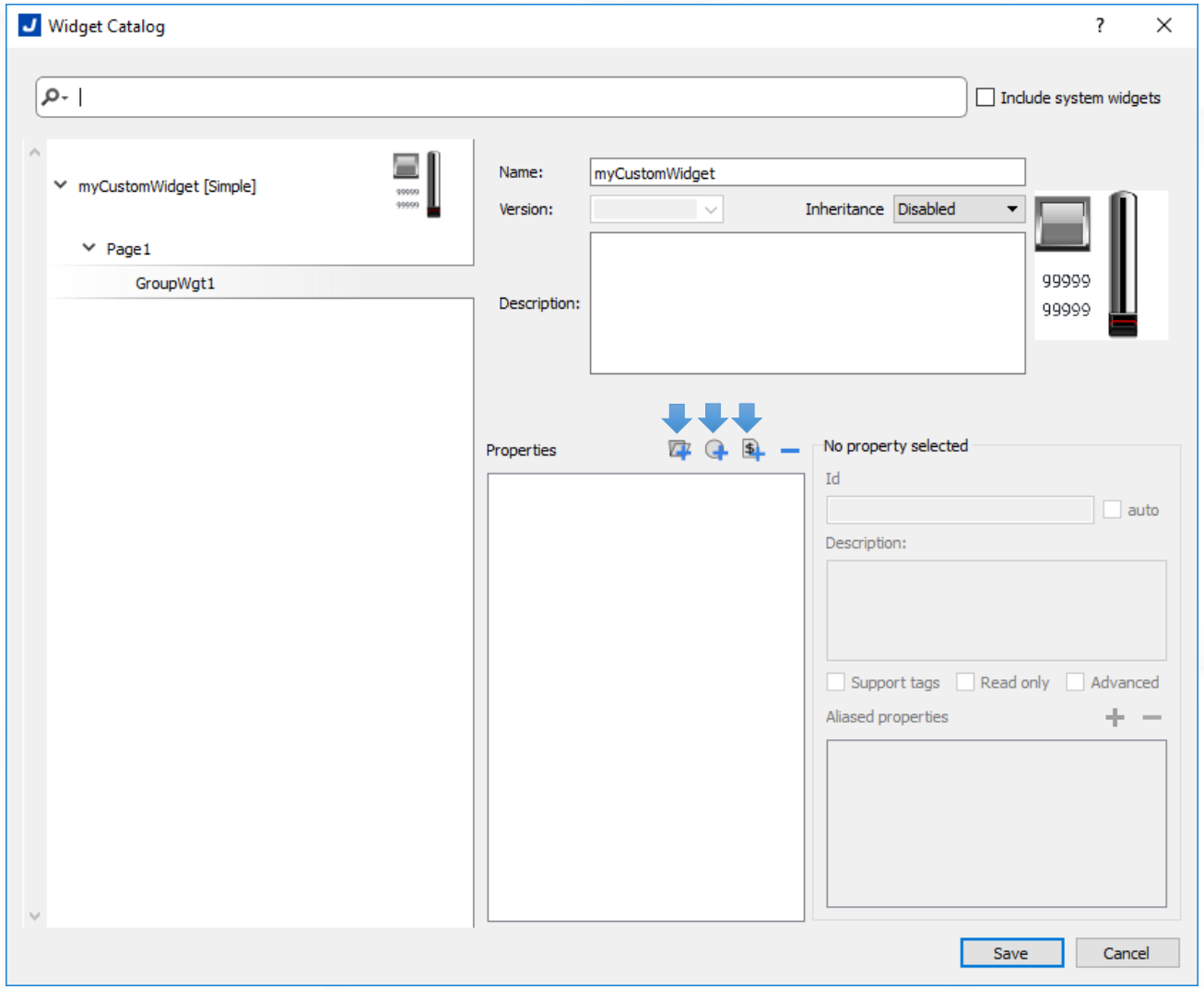
Custom Widgets

Create your own Widget Catalog

View where Custom Widgets are used in the project

Set inheritance configuration

Expose custom properties, categories and parameters



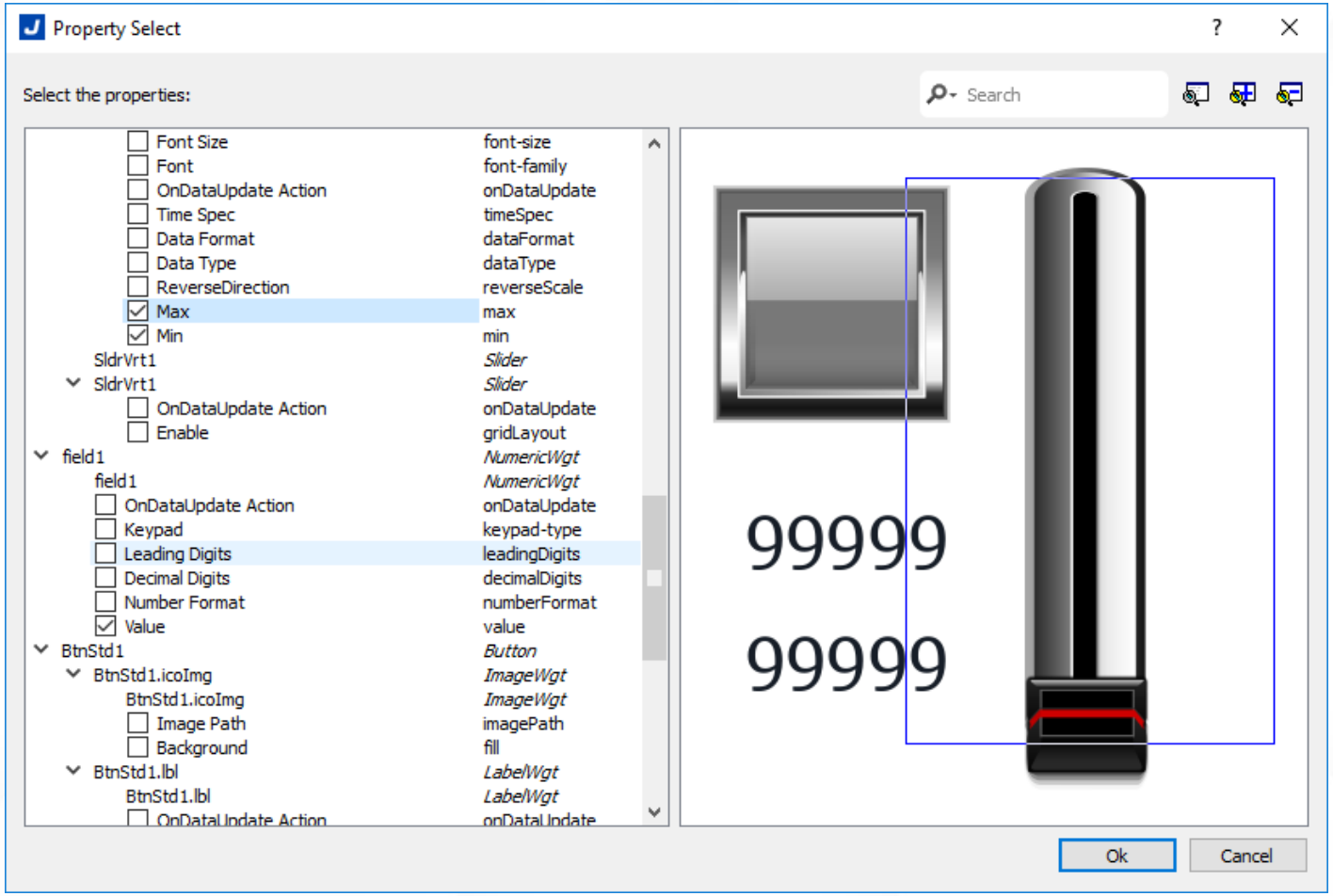
Custom Widgets

Create your own Widget Catalog

View where Custom Widgets are used in the project

Set inheritance configuration

Expose custom properties, categories and parameters



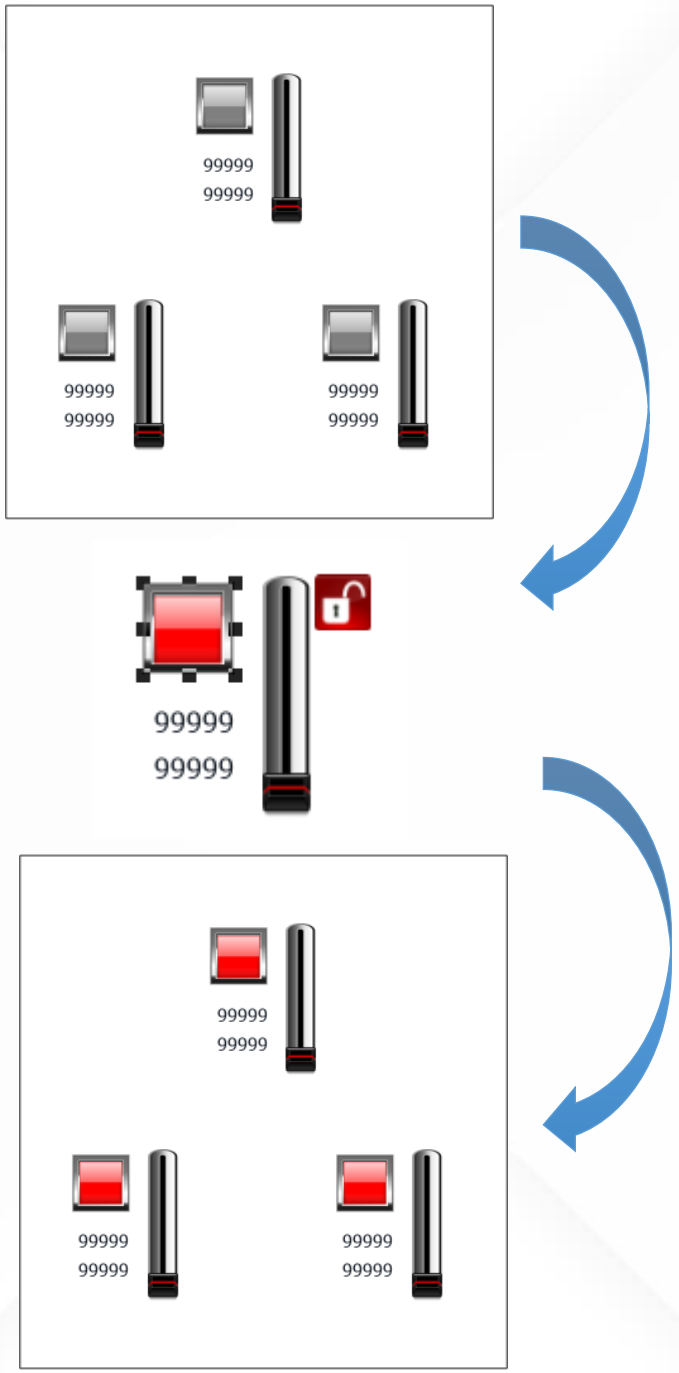
Inheritance

Inheritance allows to create a Widget model

All changes will be applied in all instances (all over the project)

Inheritance types:

- Only Graphic
- Only Logic (embedded JS code)
- Full (Graphic + Logic)



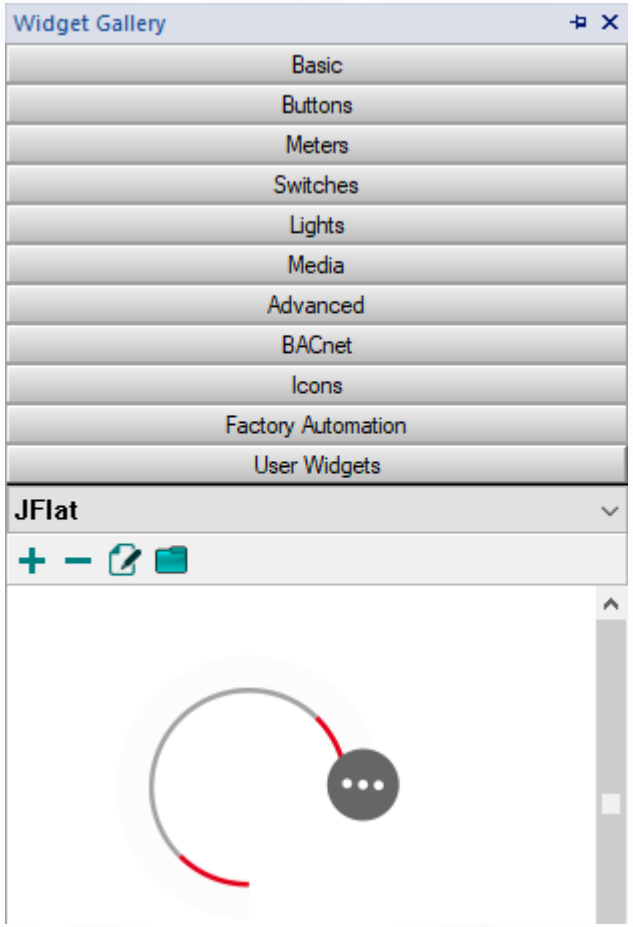
User Gallery

Inside Widget Gallery there is "User Widgets" to store custom widgets

To save a custom widget, select the edit button and copy it into “myGallery1” page

User can add/remove a personal gallery selecting [+] or [-] buttons

A personal gallery can be shared with other users Personal Gallery by default is placed under JMobile Workspace directory



CODESYS internal PLC

CODESYS as IEC 61131-3

CODESYS is a development environment for programming controller applications according to the IEC 61131-3

IEC 61131-3 is the third part of the open international standard IEC 61131 for PLC

IEC 61131-3 deals with programming languages and defines three graphical and two textual PLC programming language standards:

- Ladder diagram (LD)
- Function block diagram (FBD)
- Structured text (ST)
- Instruction list (IL)
- Sequential function chart (SFC)

CODESYS V2.3 and CODESYS V3.5

CODESYS V 3.5 internal PLC available for all HMI models

CODESYS V2.3 internal PLC available only for eTOP500/600 WCE based HMI models

Programming Languages

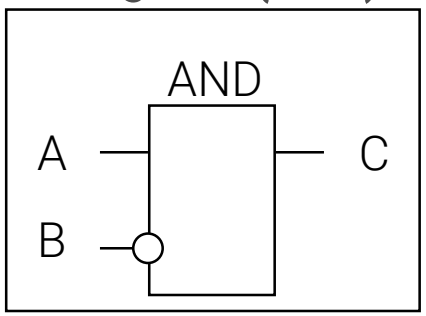
Instruction List (IL)

```
LD      A
ANDN    B
ST      C
```

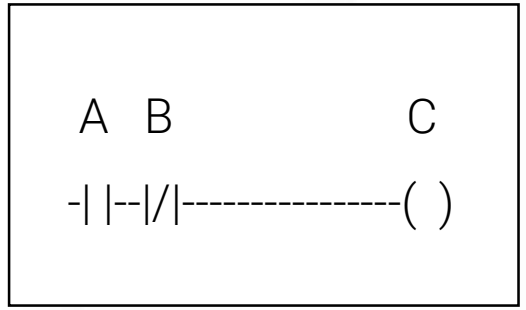
Structured Text (ST)

```
C:= A  AND  NOT  B
```

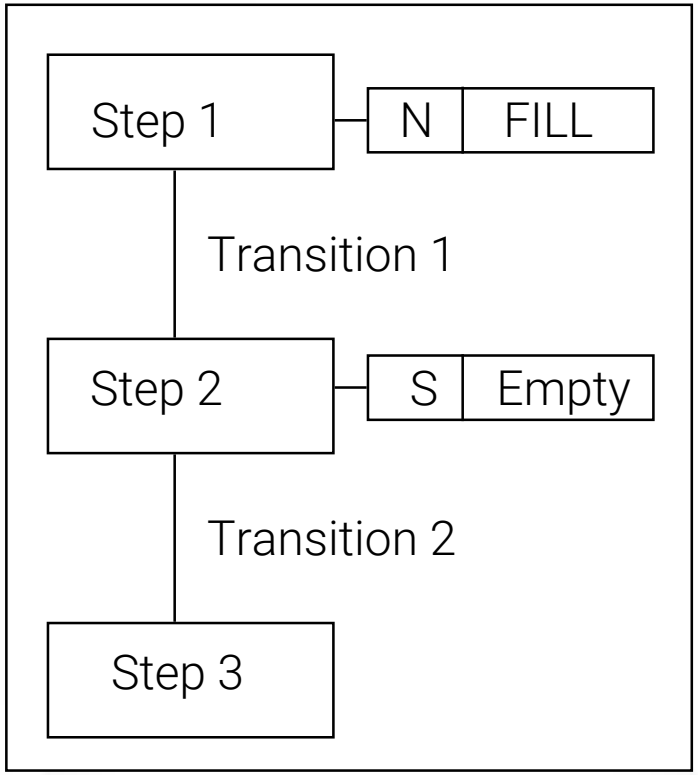
Function Block Diagram (FBD)



Ladder Diagram (LD)



Sequential Function Chart (SFC)



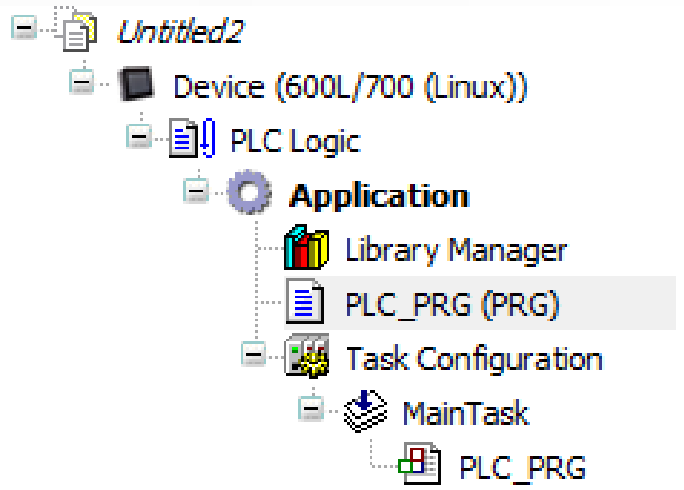
IEC 61131-3 Data Types

IEC 61131-3 Data Type	CODESYS Data Type	JMobile Data Type	Limits	Memory Space
BOOL	BOOL	Boolean	0 ... 1	1 bit data
SINT	SINT	Byte	-128 ... 127	8-bit data
USINT	USINT / BYTE	UnsignedByte	0 ... 255	8-bit data
INT	INT	Short	-32768 ... 32767	16-bit data
UINT	UINT / WORD	UnsignedShort	0 ... 65535	16-bit data
DINT	DINT	Int	-2.1e9 ... 2.1e9	32-bit data
UDINT	UDINT / DWORD	UnsignedInt	0 ... 4.2e9	32-bit data
LINT	---	---	-9.2e19 ... 0.2e19	64-bit data
REAL	REAL	Float	1.17e-38 ... 3.40e38	IEEE single-precision 32-bit floating point type
LREAL	---	Double	2.2e-308 ... 1.79e308	IEEE double-precision 64-bit floating point type

CODESYS objects

POUs (Program Organization Units)

default: PLC_PRG
can be Program, Function or Function block
every POU has variable declaration and code
language can be selected



Library Manager

standard 3S libraries for utilities
EXOR libraries to manage specific scenarios
PT100 and thermocouples (PLIO03)
NM2000

NEW
in 4.0

Communication stacks

CANOpen Master

Modbus TCP/RTU Master/Slave

EtherNet/IP Scanner

Profinet I/O Master

EtherCAT Master

PowerLink Master

NOTE: Support depends on hardware capabilities

CODESYS Development software

Download from EXOR website

Many versions can be installed in the same PC

Standard 3S Environment

exorint.com/en/software/jmobile#downloads

CODESYS internal PLC Downloads

CODESYS 3.5 (LAST RELEASED)
CODESYS v3 (OLD VERSIONS)
CODESYS v2.3

NOTE: CODESYS v2.3 is not compatible with CODESYS v3.5

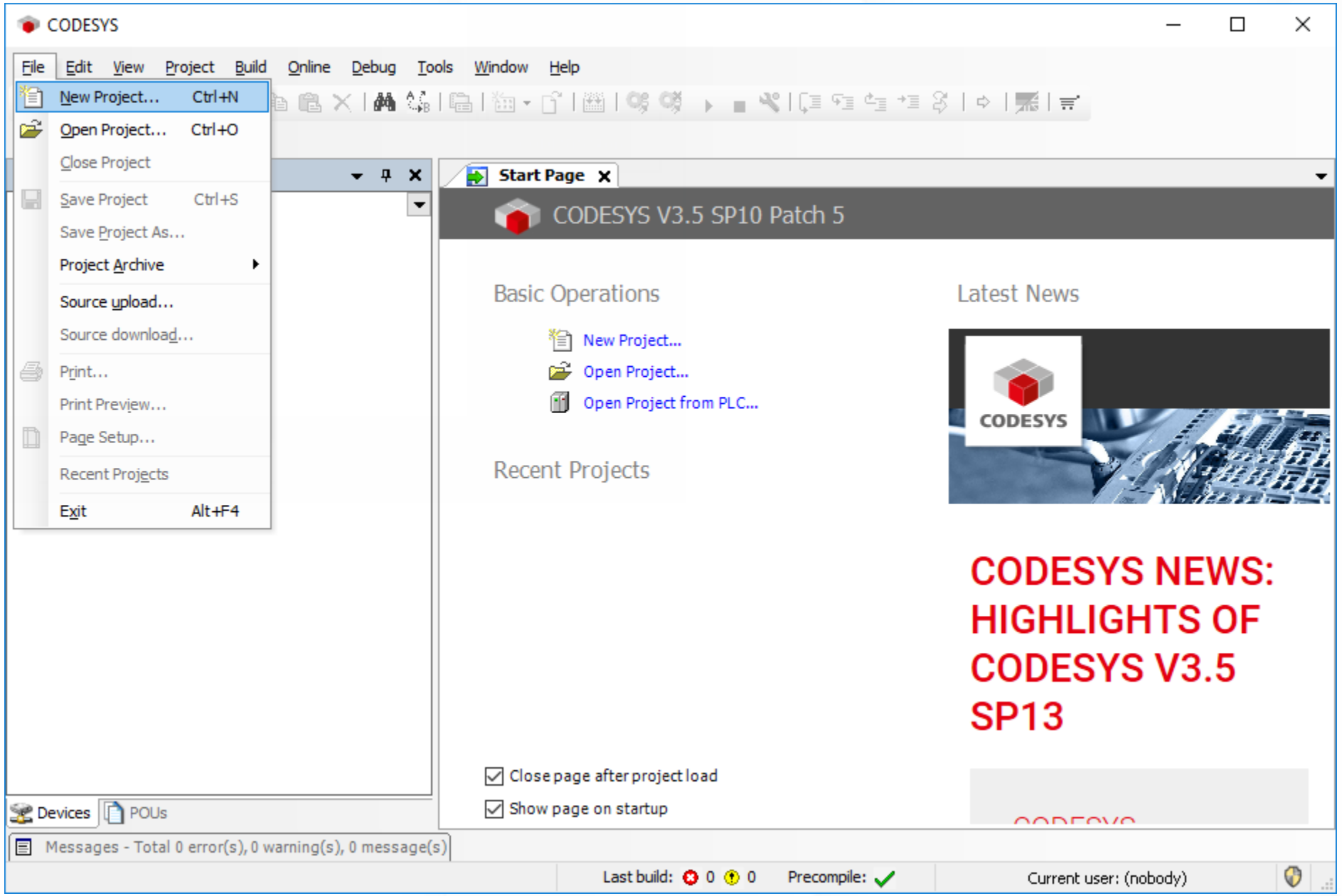
CODESYS EXOR Targets

Distributed into JMobile Installation folder

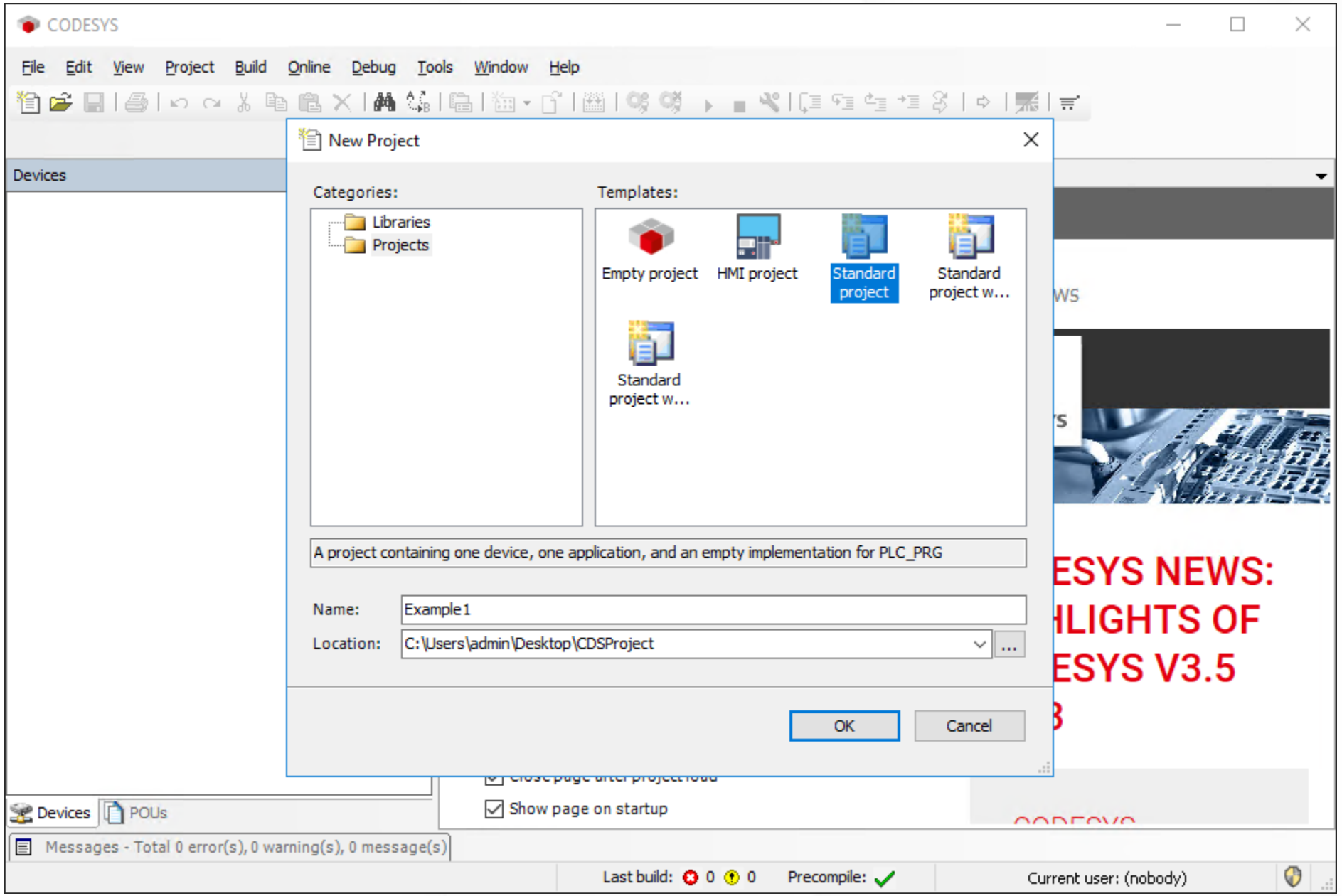
Once CODESYS is installed, double click on Target Package file
to install EXOR Targets into CODESYS software

Or by opening **Tools > Package Manager > Install** and selecting the package

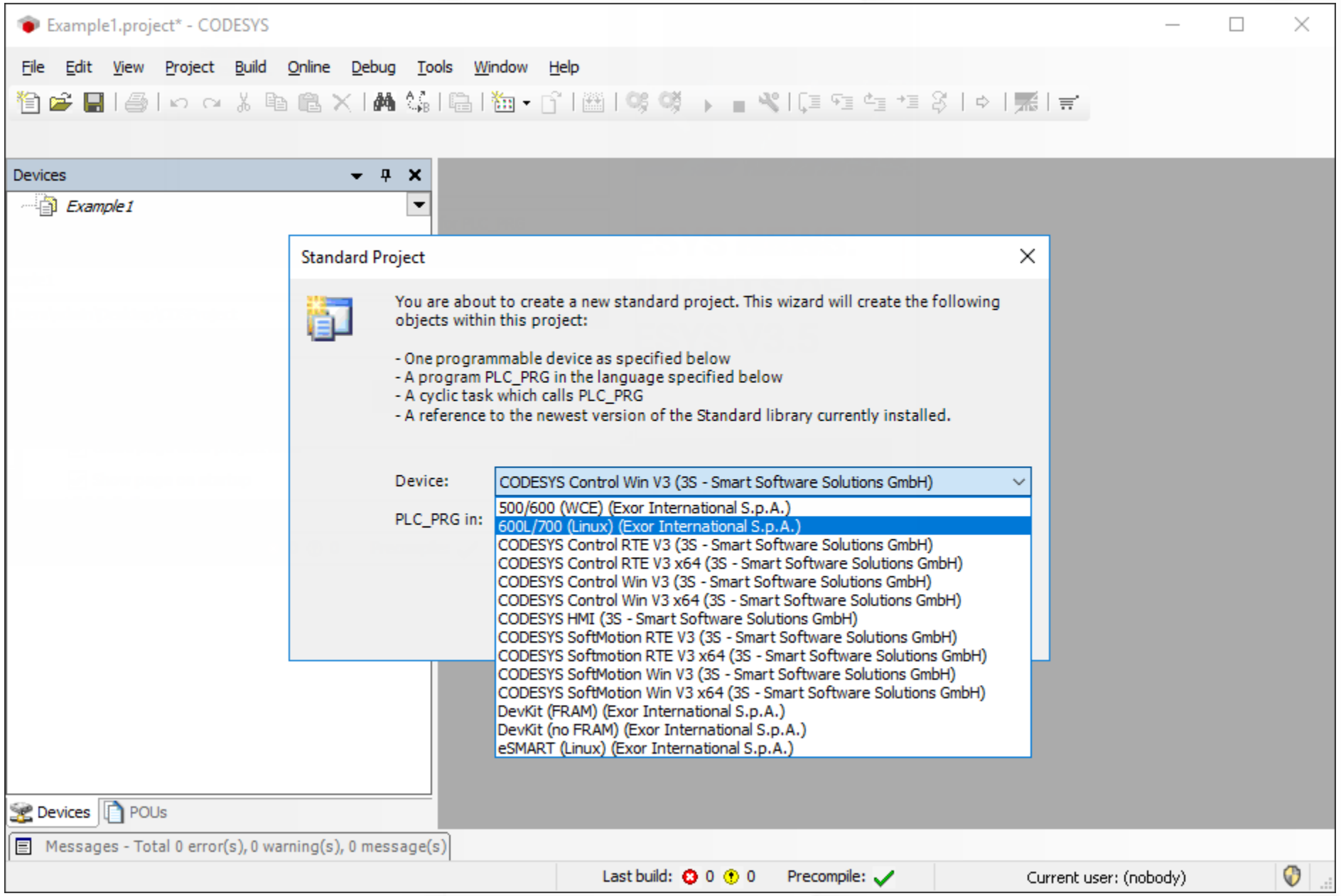
CODESYS v3: create a new project



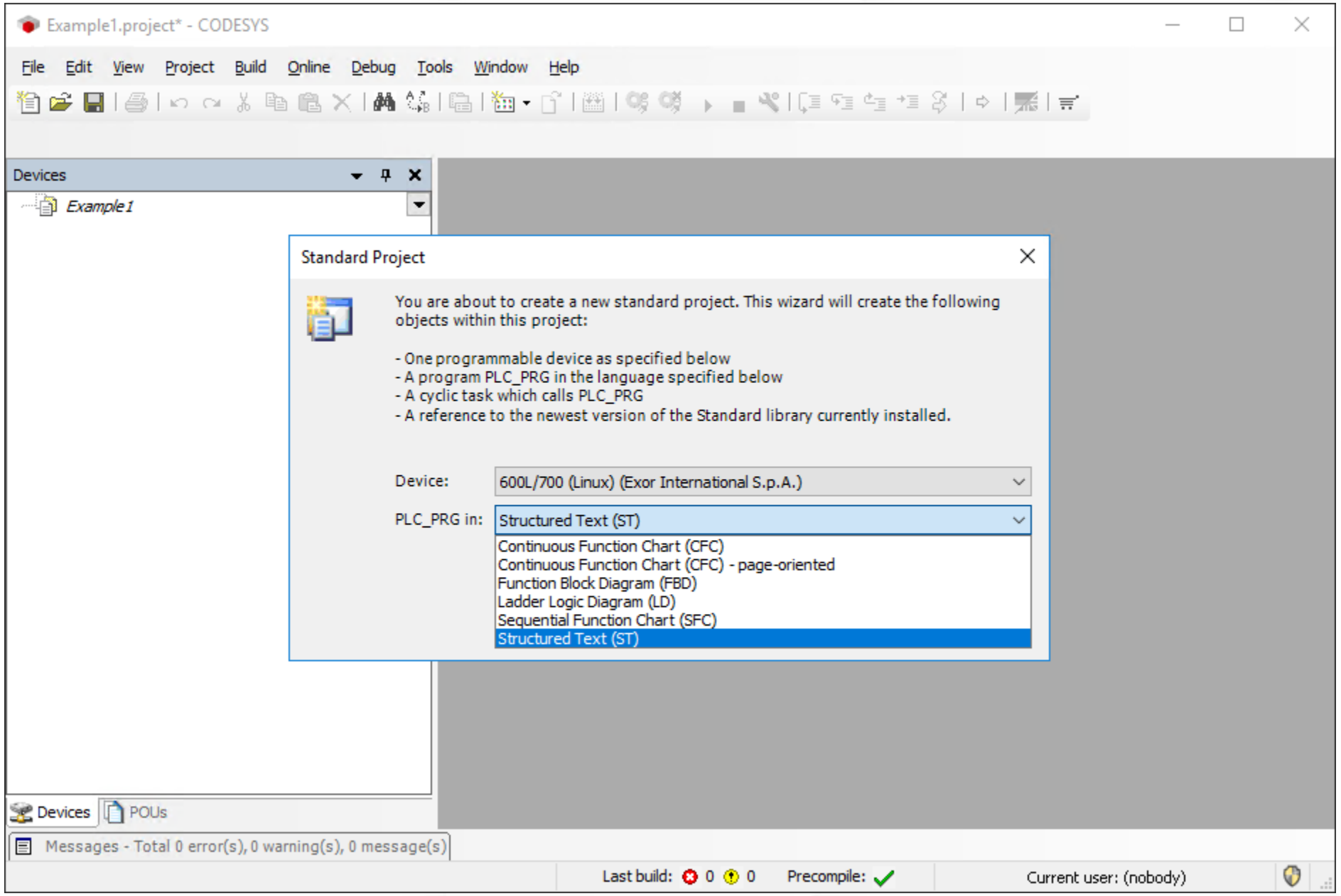
CODESYS v3: create a new project



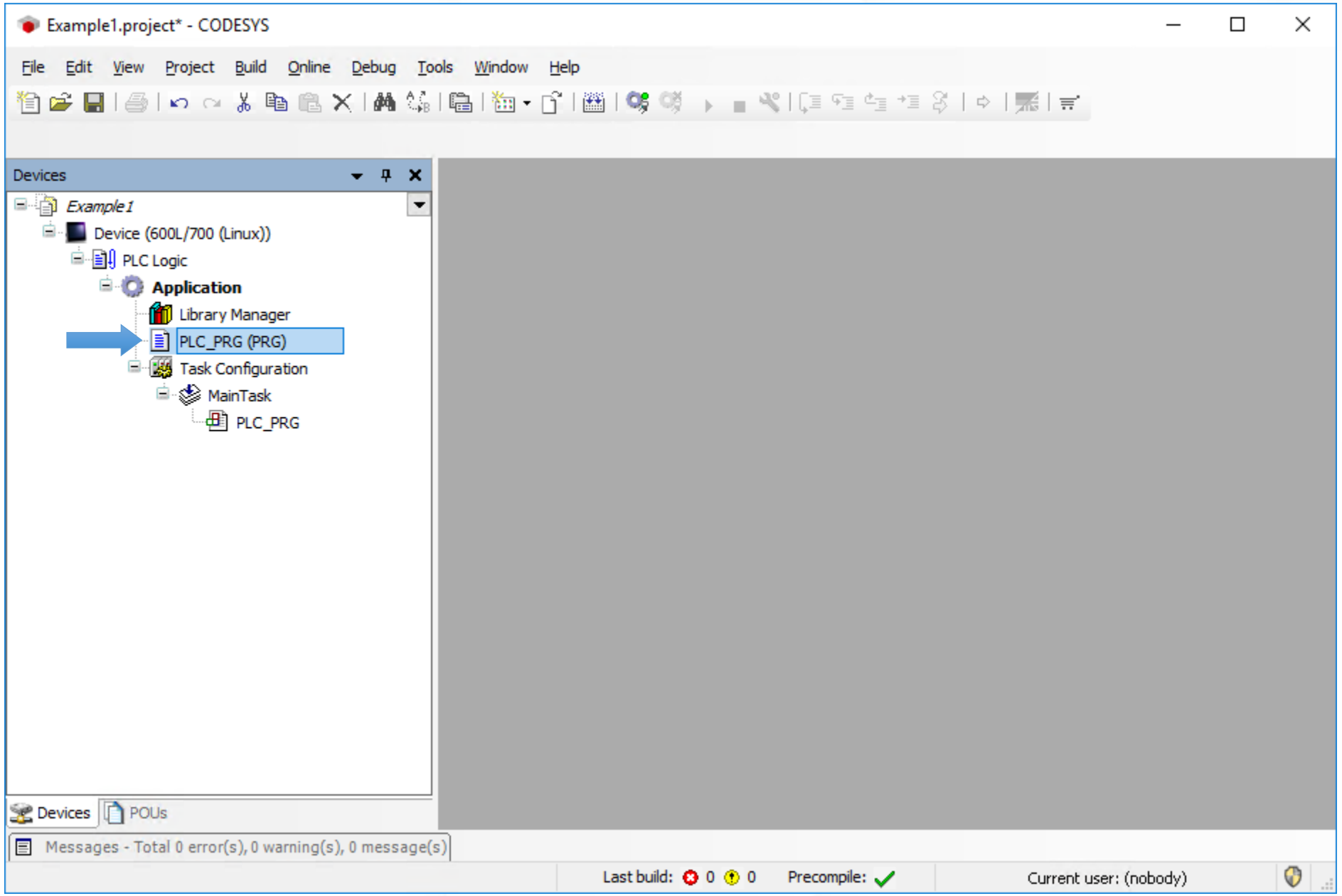
CODESYS v3: create a new project



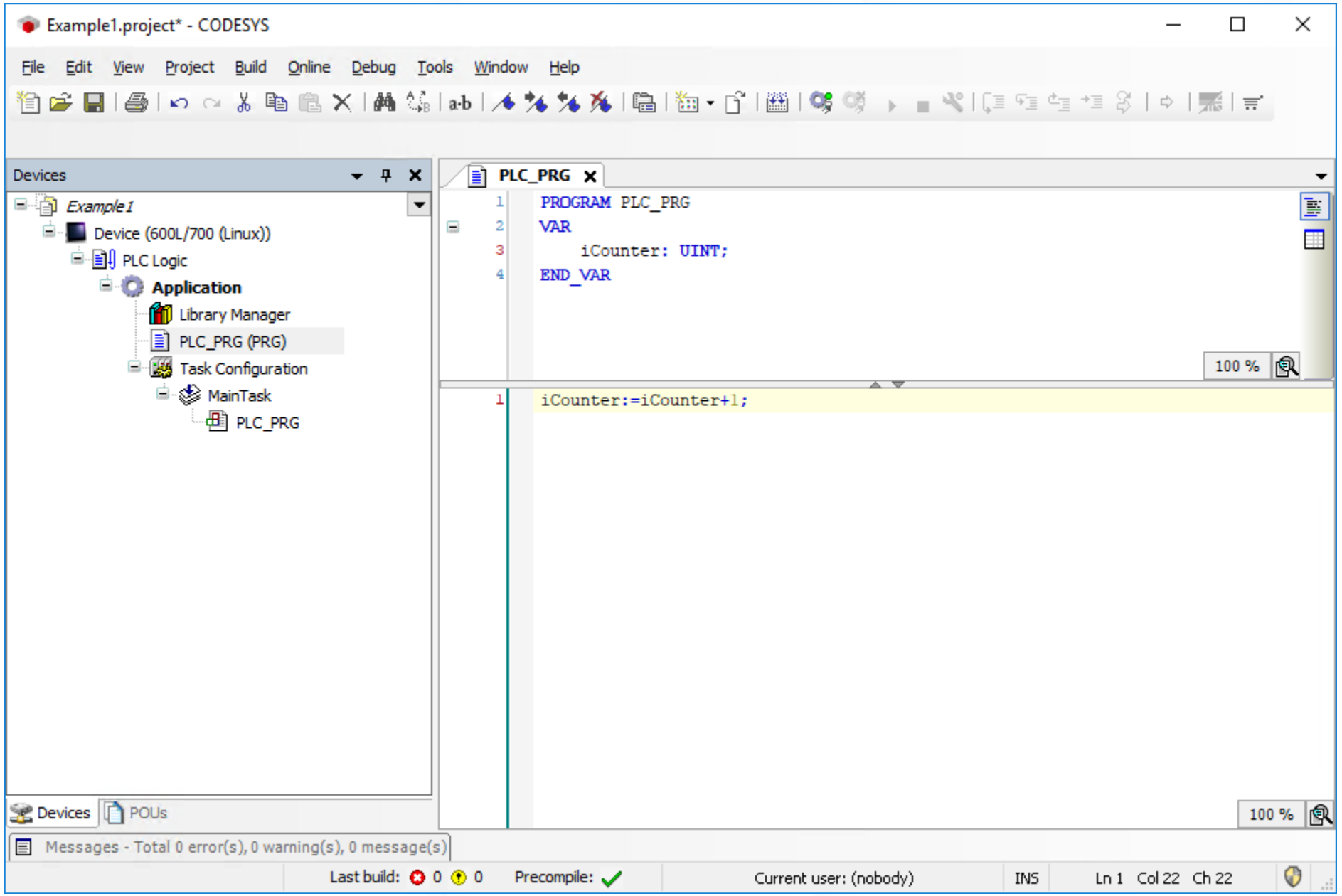
CODESYS v3: create a new project



CODESYS v3: create a new project



CODESYS v3: create a new project



Create a new project - remarks

Select the device based on HMI series:

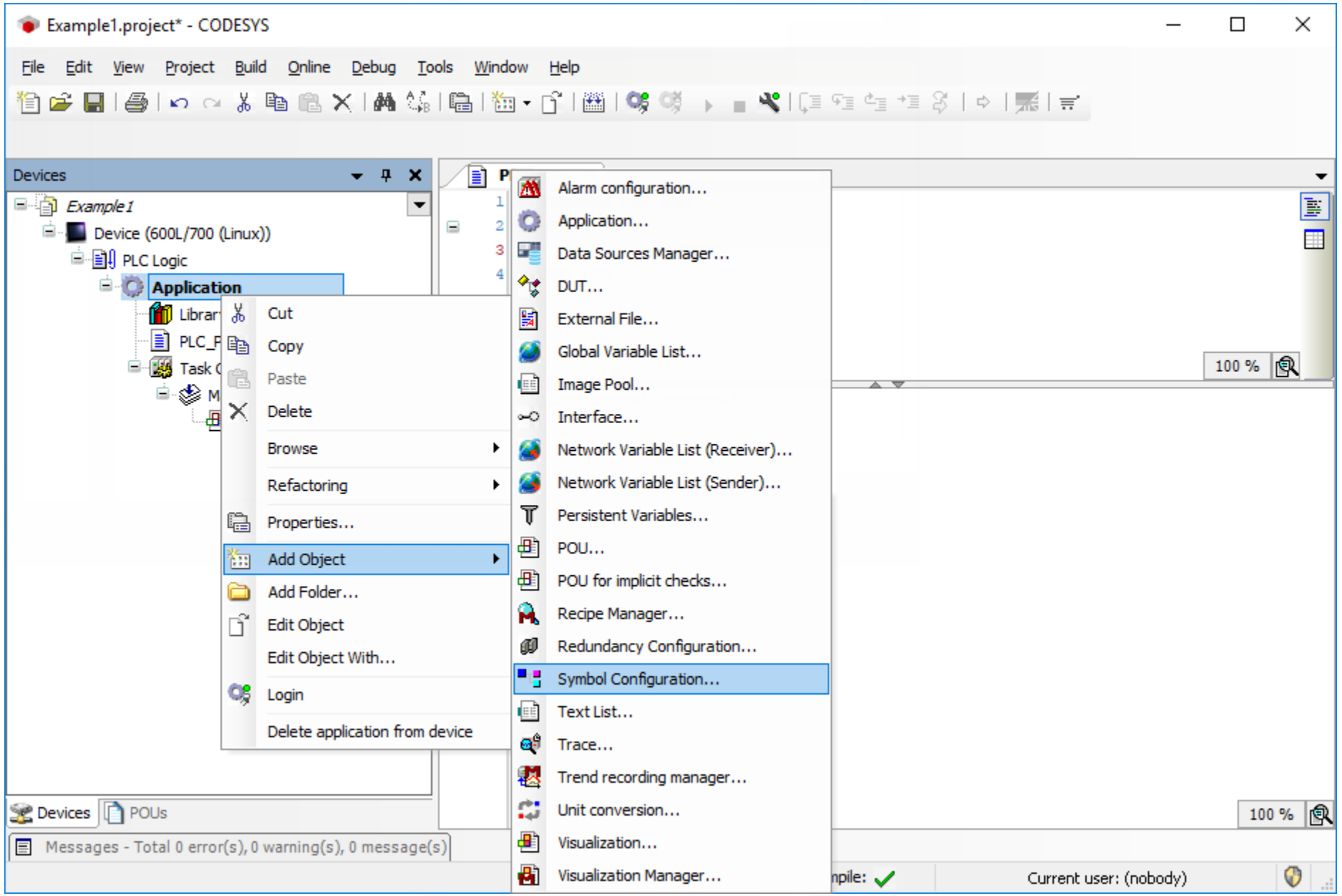
500/600 (WCE)
for eTOP500 and eTOP600

600L/700 (Linux)
for JSmart700, eX700, eXware700, eSMART107

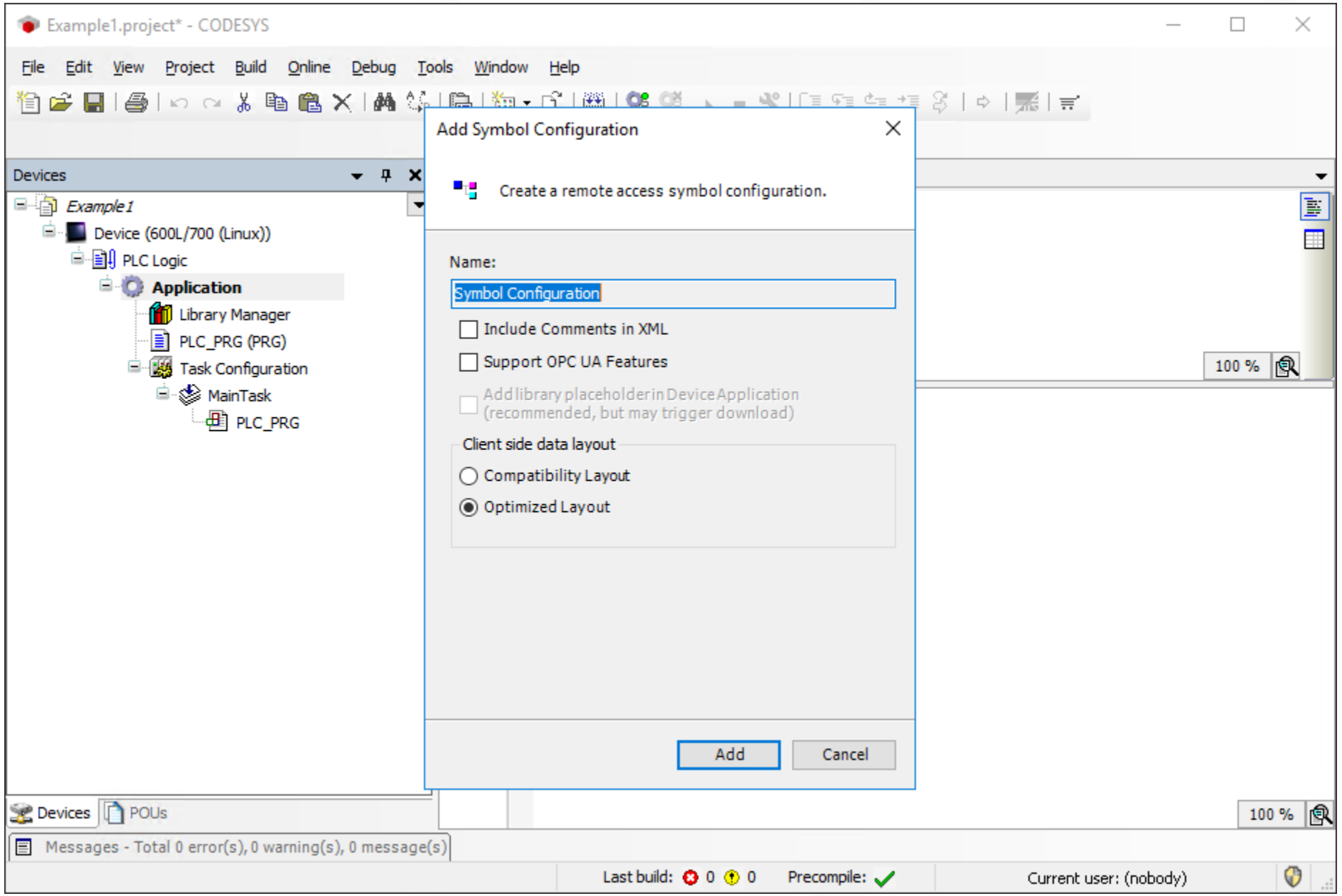
eSMART (Linux) *
for eSMART04, 07M, 10

* CODESYS Retentive variables not supported

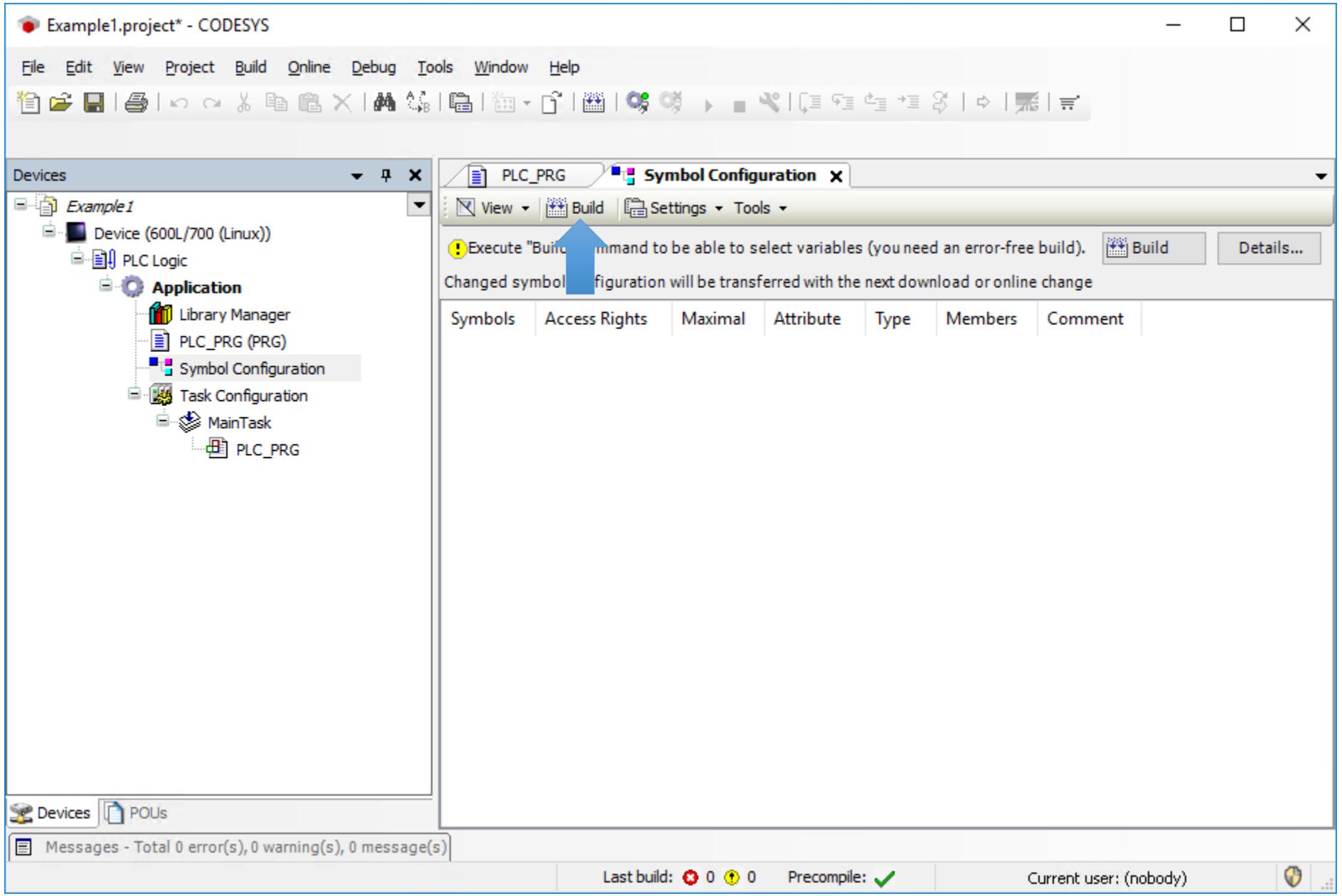
CODESYS v3: symbol configuration



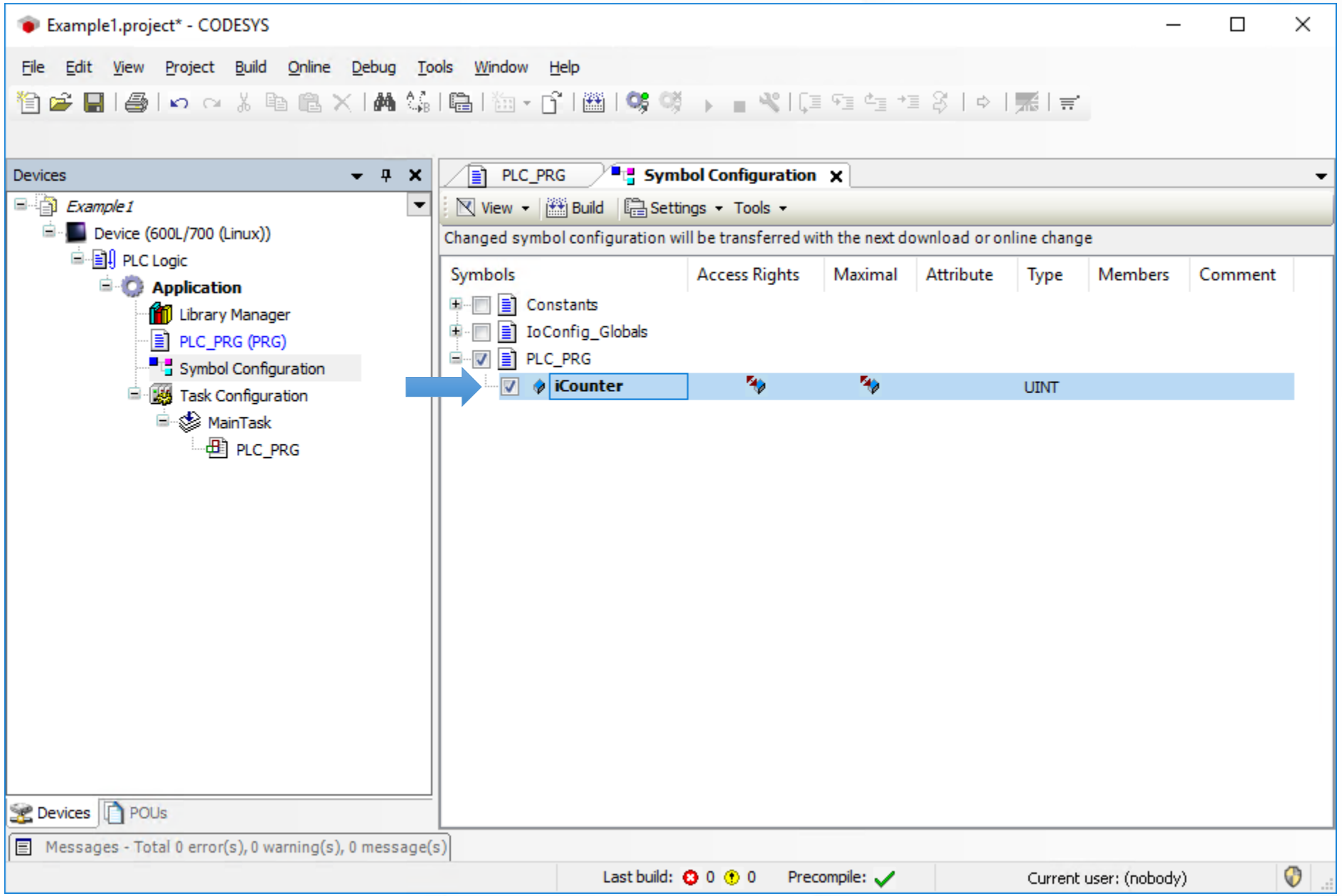
CODESYS v3: symbol configuration



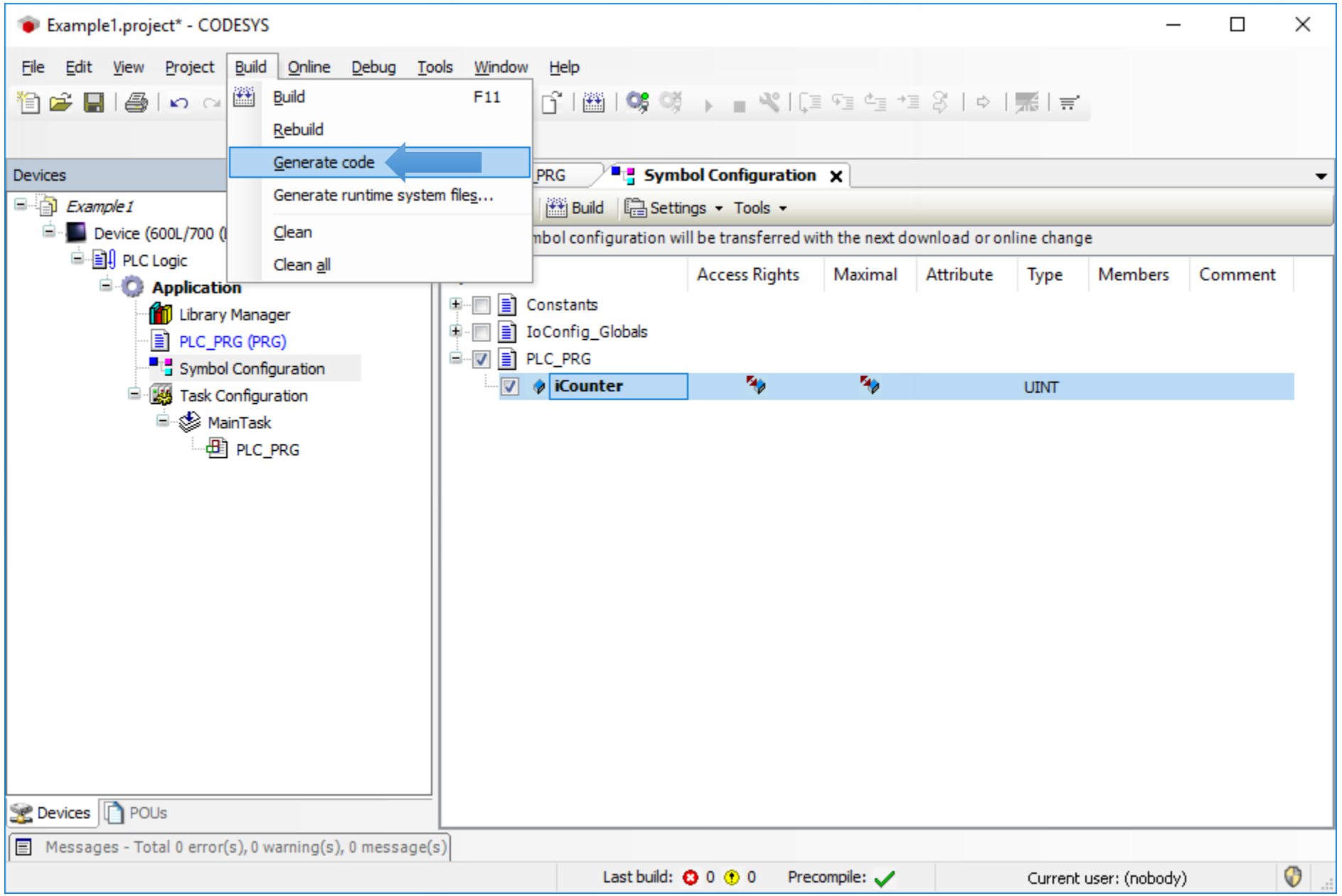
CODESYS v3: symbol configuration



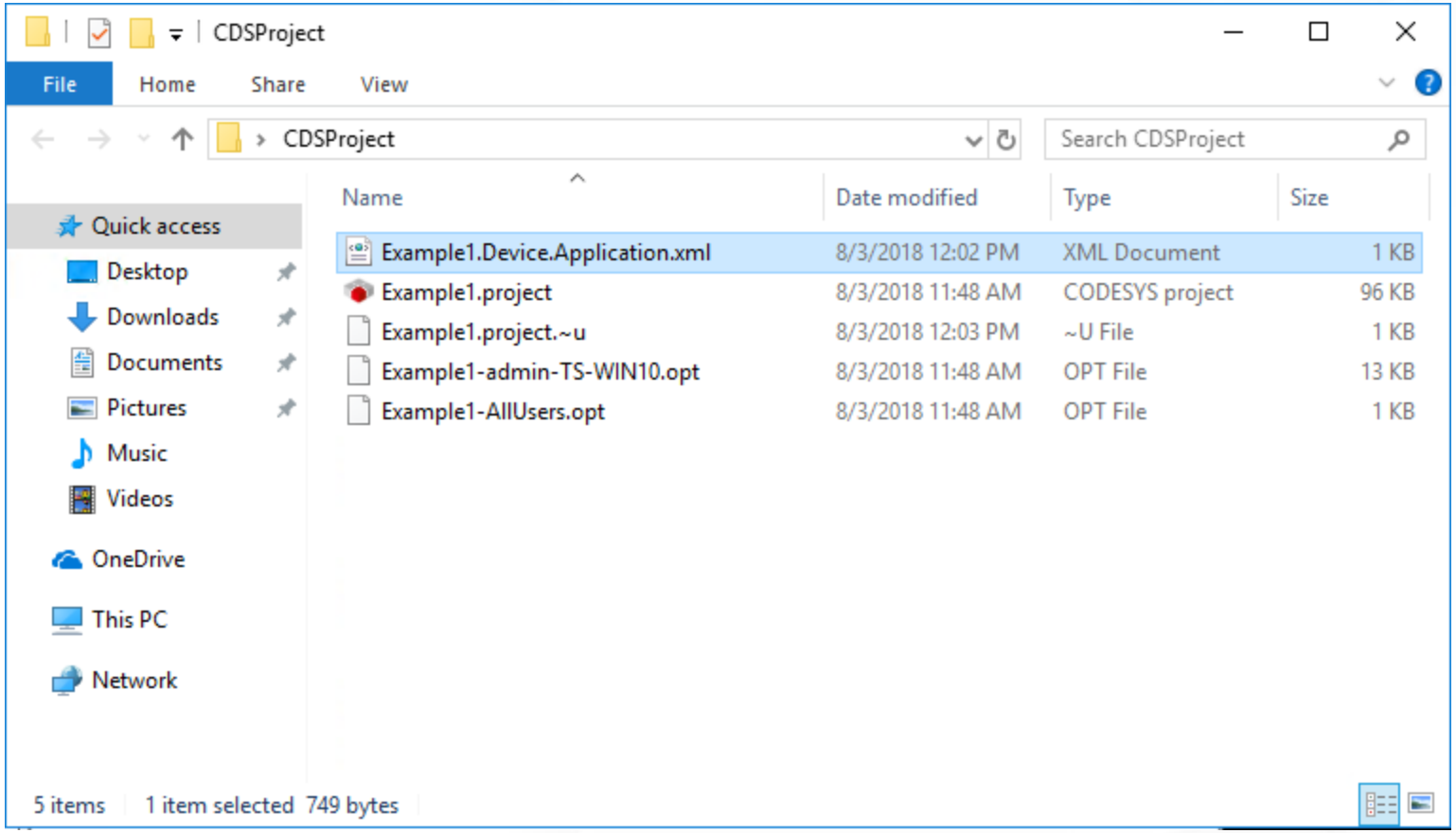
CODESYS v3: symbol configuration



CODESYS v3: symbol configuration



CODESYS v3: symbol configuration



Symbol configuration - remarks

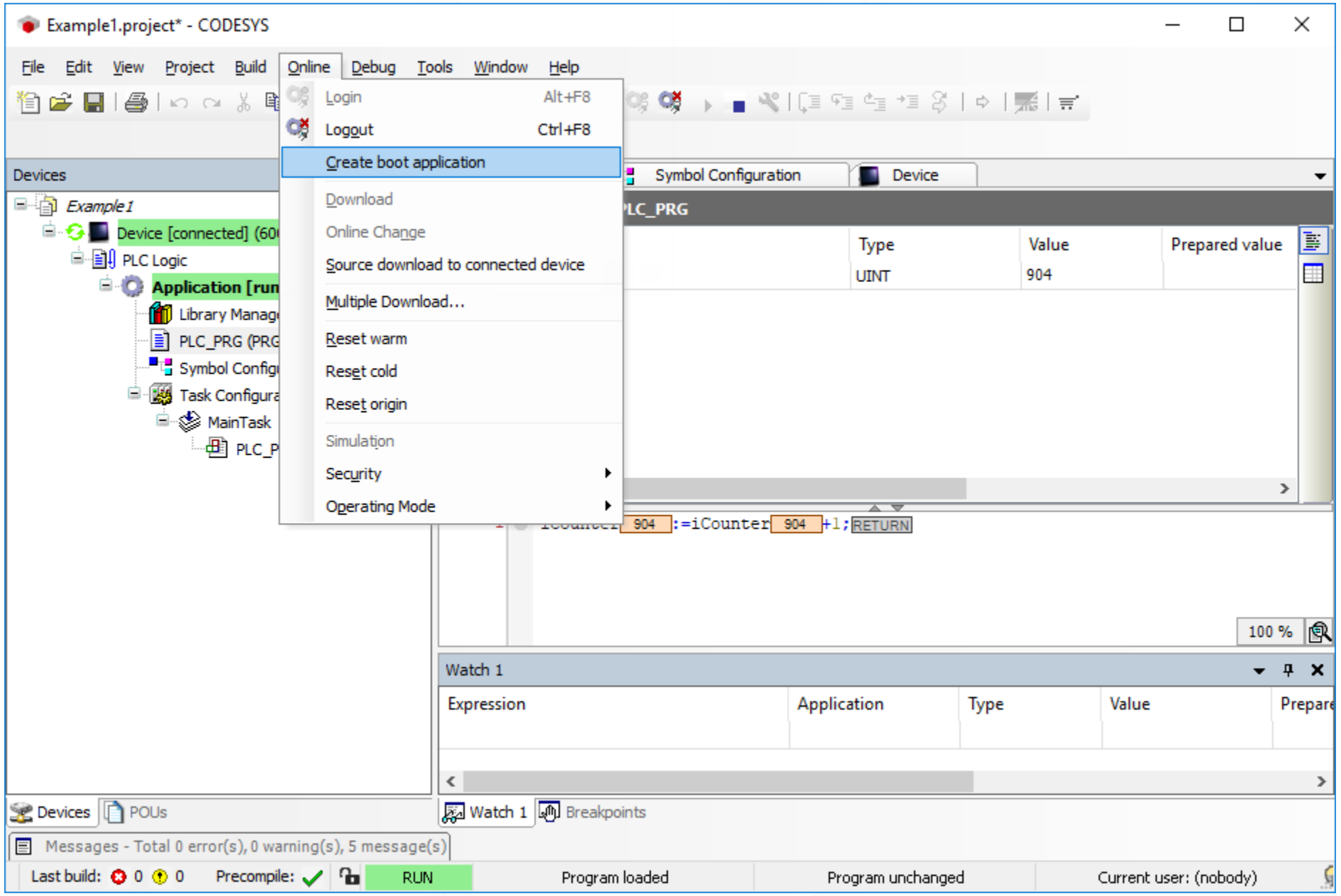
Right-Click on Application, then select
Add Object > Symbol Configuration...

Inside Symbol Configuration click "Build"

Go to Build > Generate code to generate the .xml file
xml file is used as the tag import file to bring all selected variables into the JMobile project

OPTIONAL: select "Support OPC UA features" to expose symbols via CODESYS OPC UA Server

CODESYS v3: download to PLC



Download to PLC - remarks

Online > Login

will open a dialog box to confirm the download to the PLC. At that time, the code is in RAM. A restart of HMI will erase the PLC code

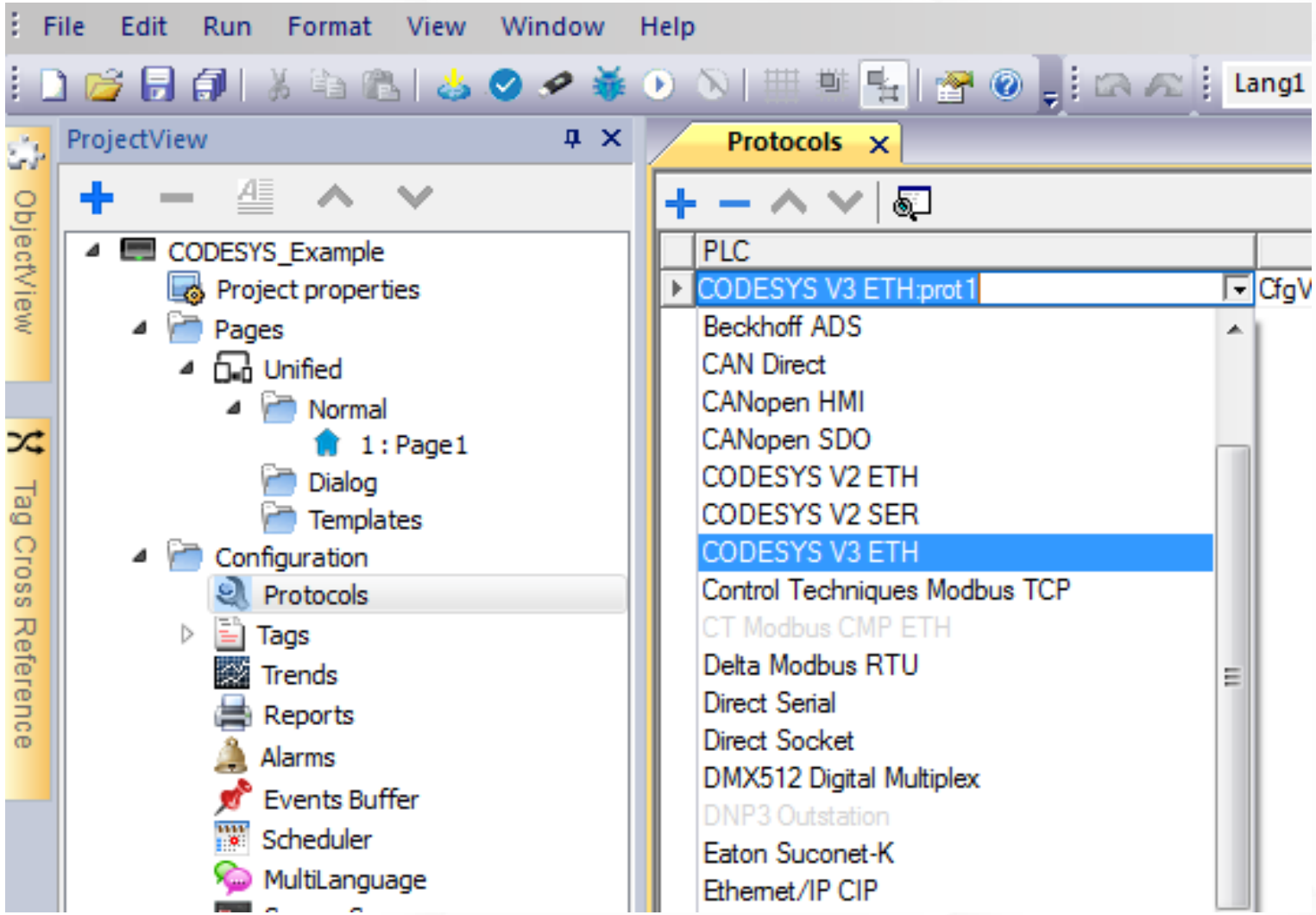
Online > Create boot application

will save the PLC application to flash

Debug > Start

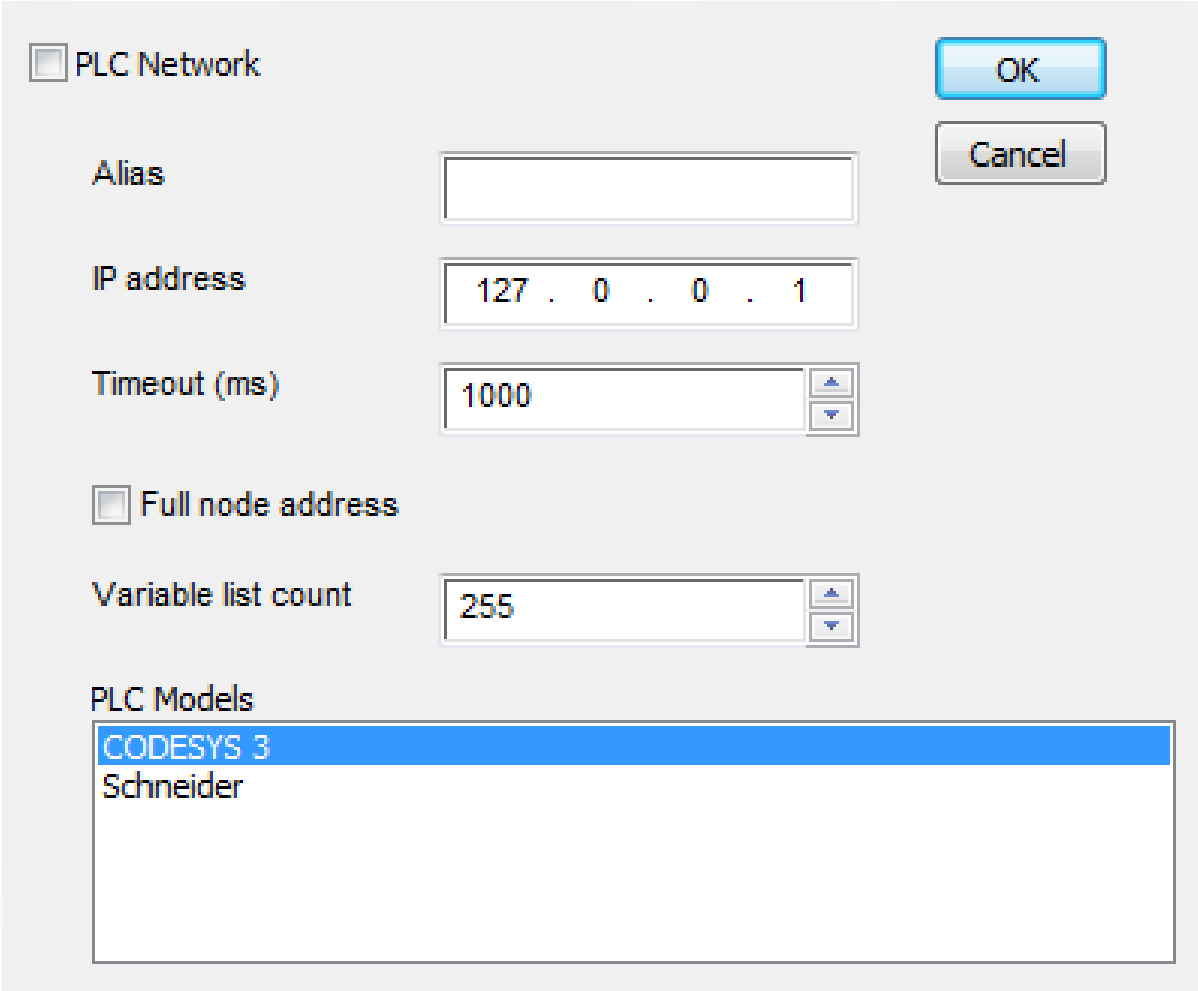
will put the PLC in Run mode

CODESYS v3: configure protocol in JMobile



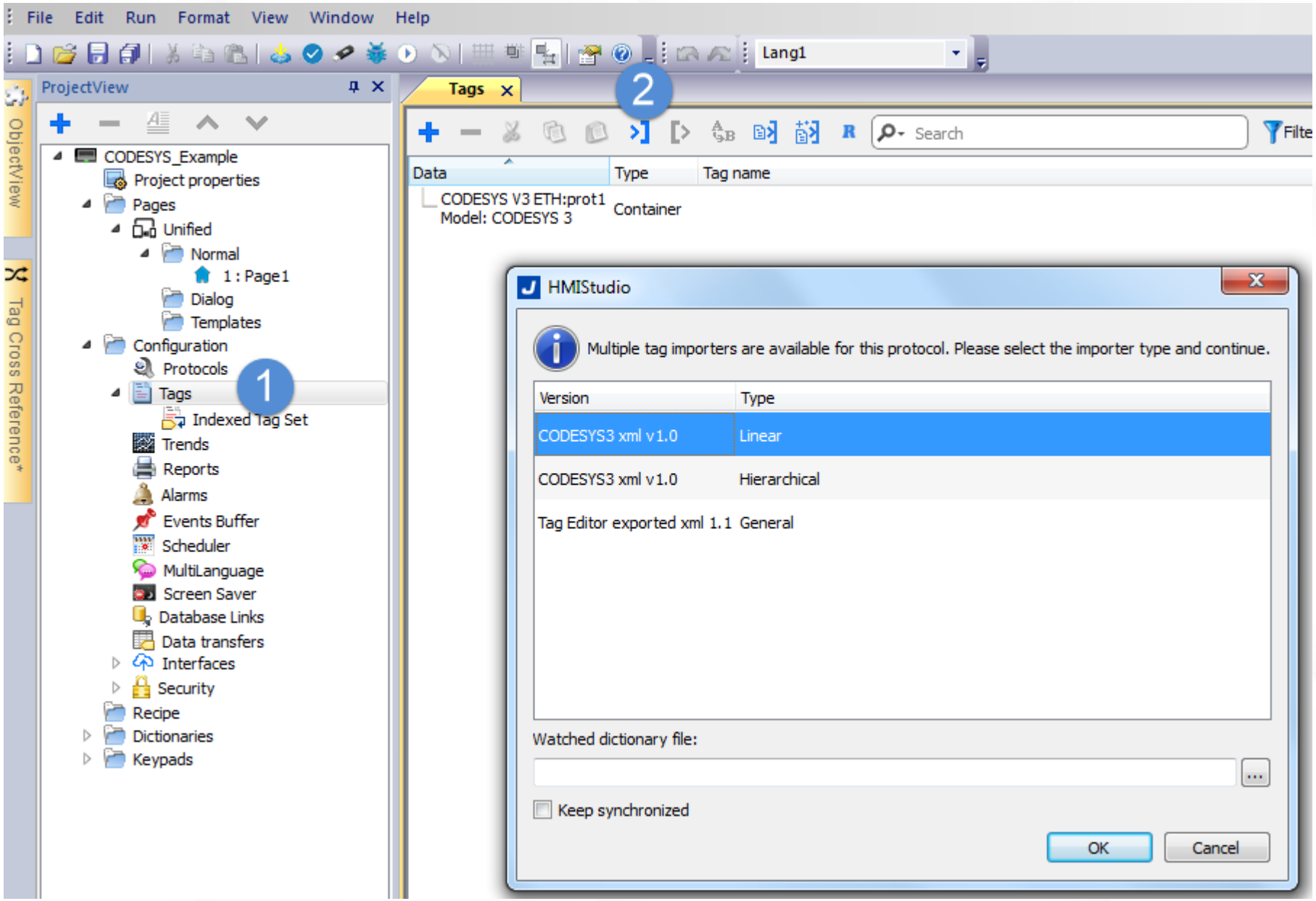
CODESYS v3: configure protocol in JMobile

Set localhost IP Address in protocol configuration (127.0.0.1)



CODESYS v3: configure protocol in JMobile

Import XML file containing CODESYS variables



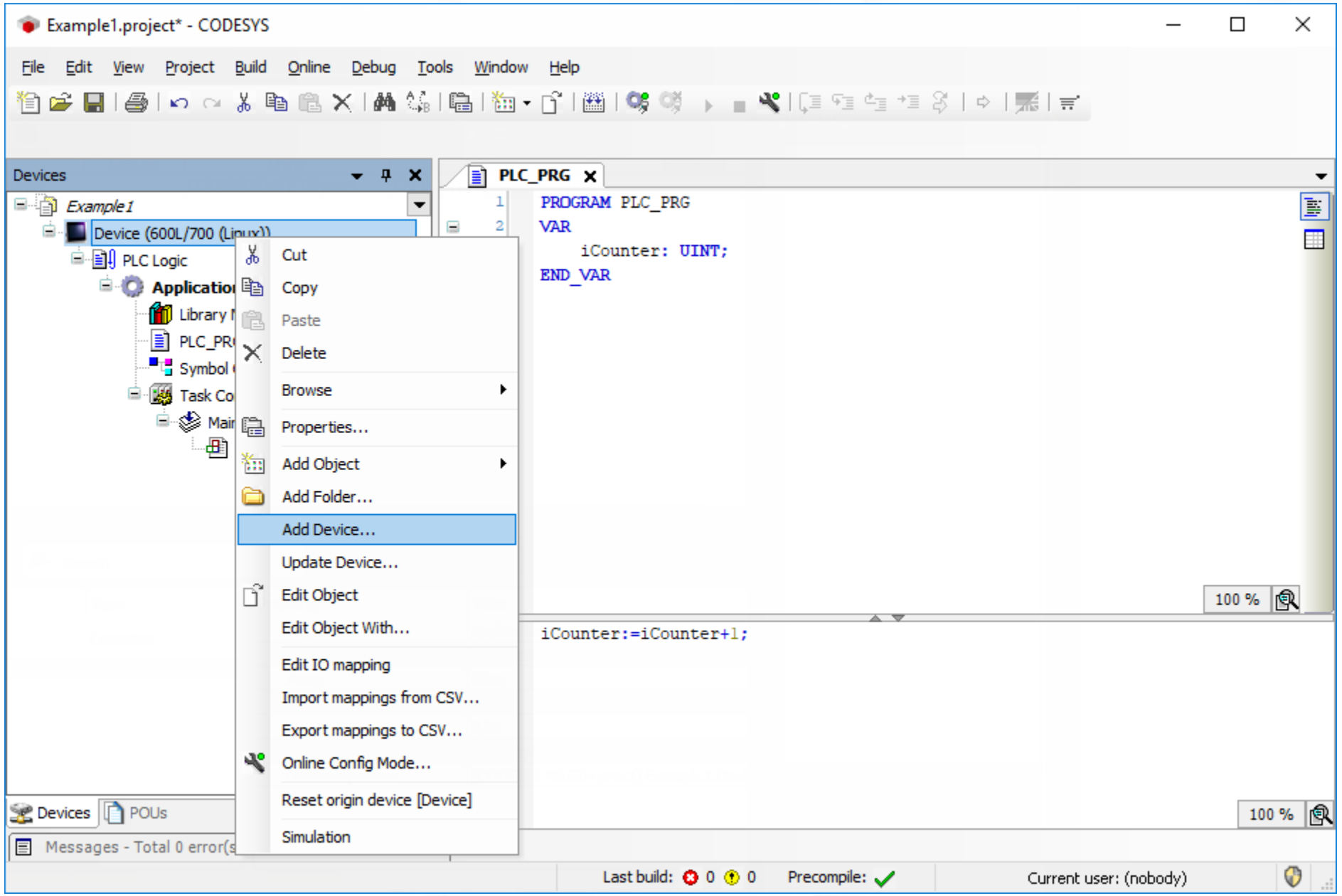
Configure protocol - remarks

Add CODESYS v3 ETH protocol

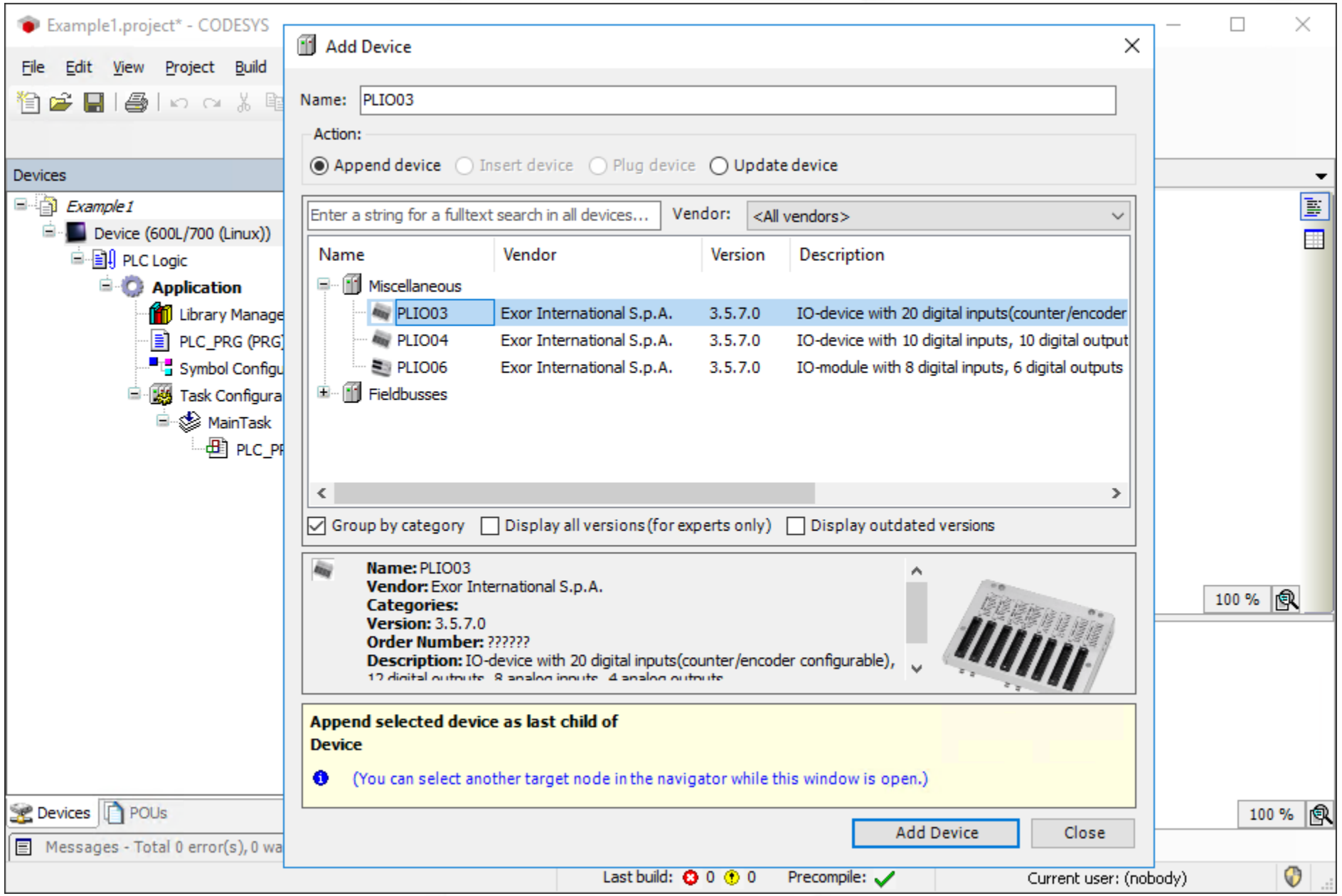
Configure PLC IP address as 127.0.0.1

Import Tags from symbol file ".xml"

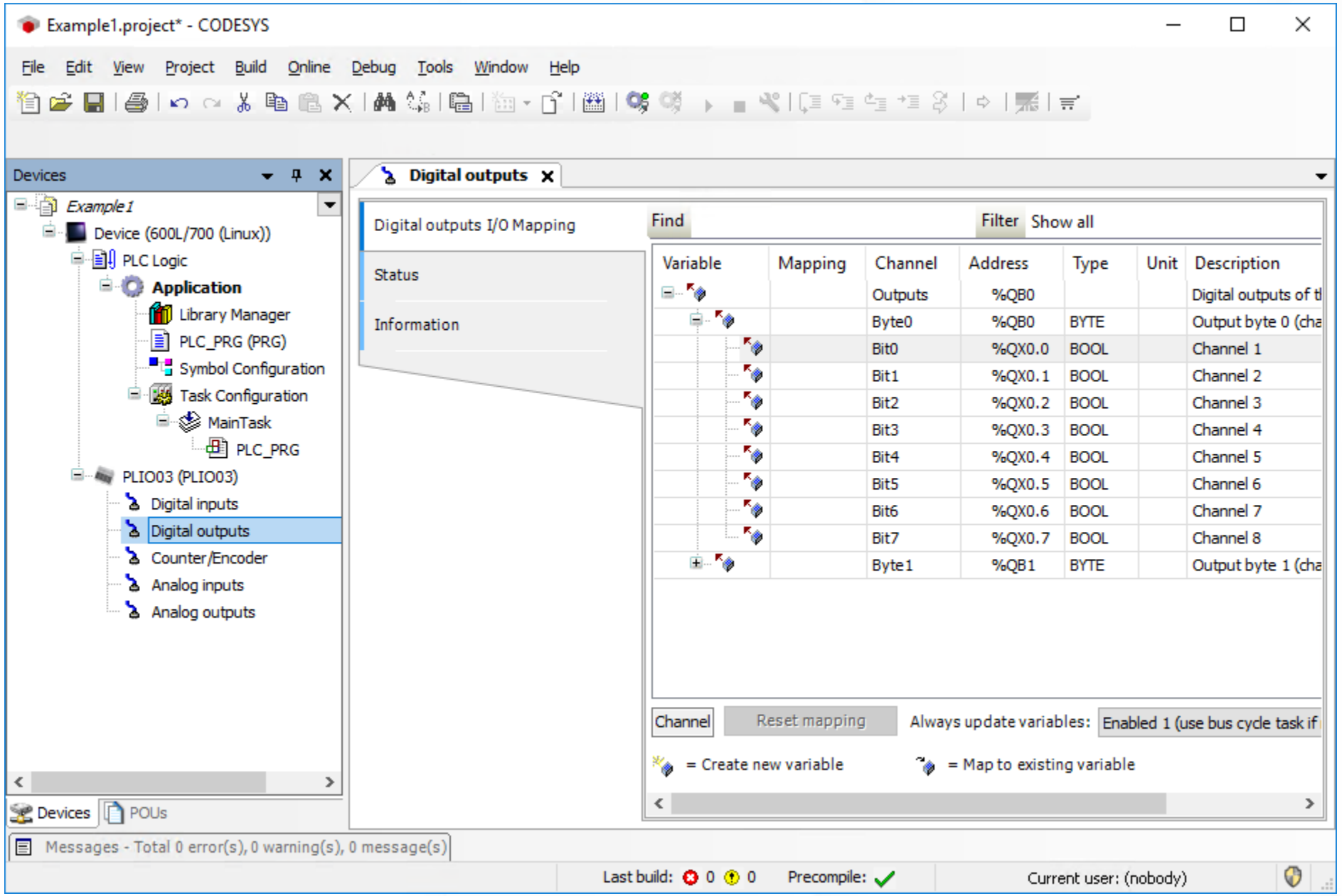
PLC Configuration: add local I/O plug-in



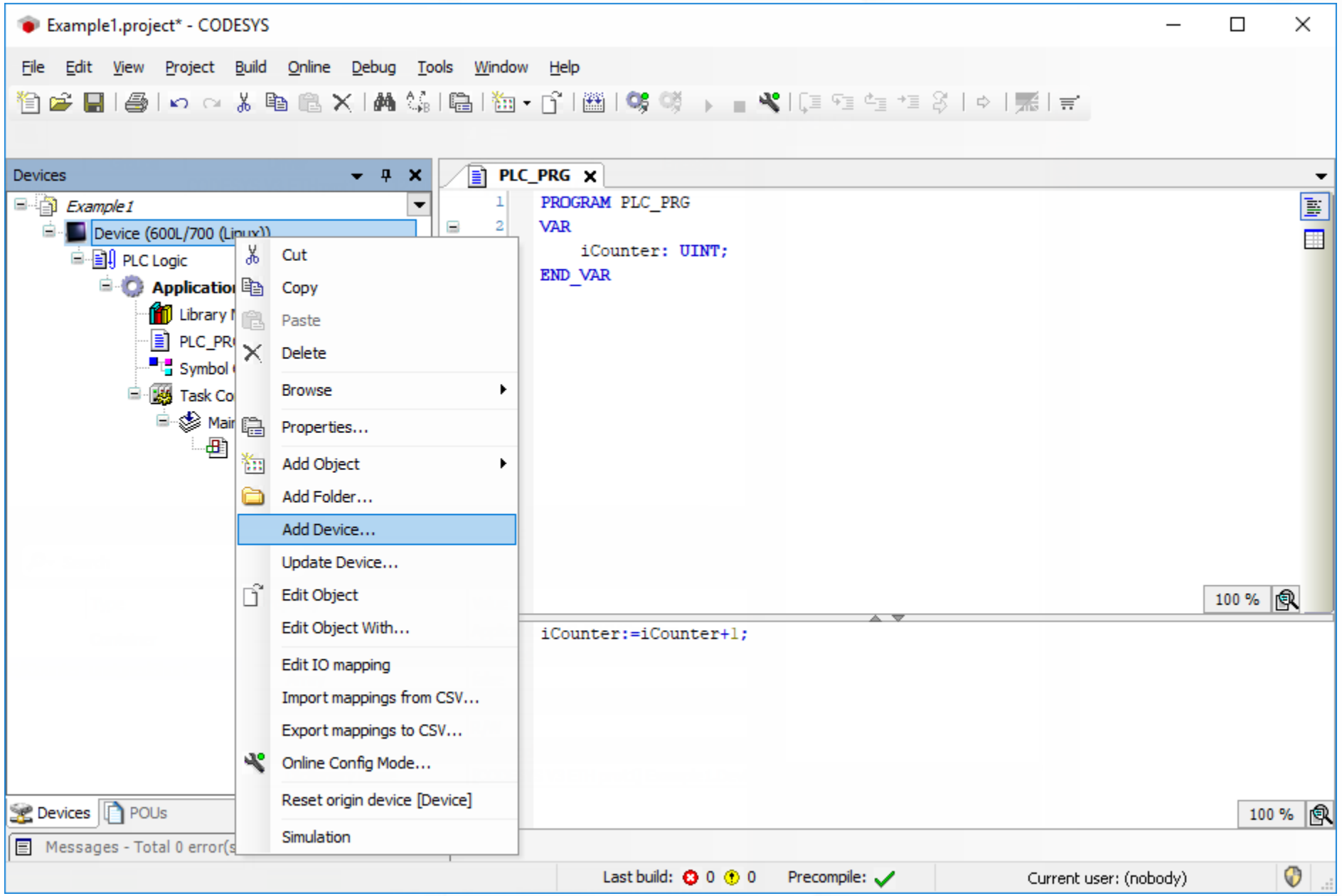
PLC Configuration: add local I/O plug-in



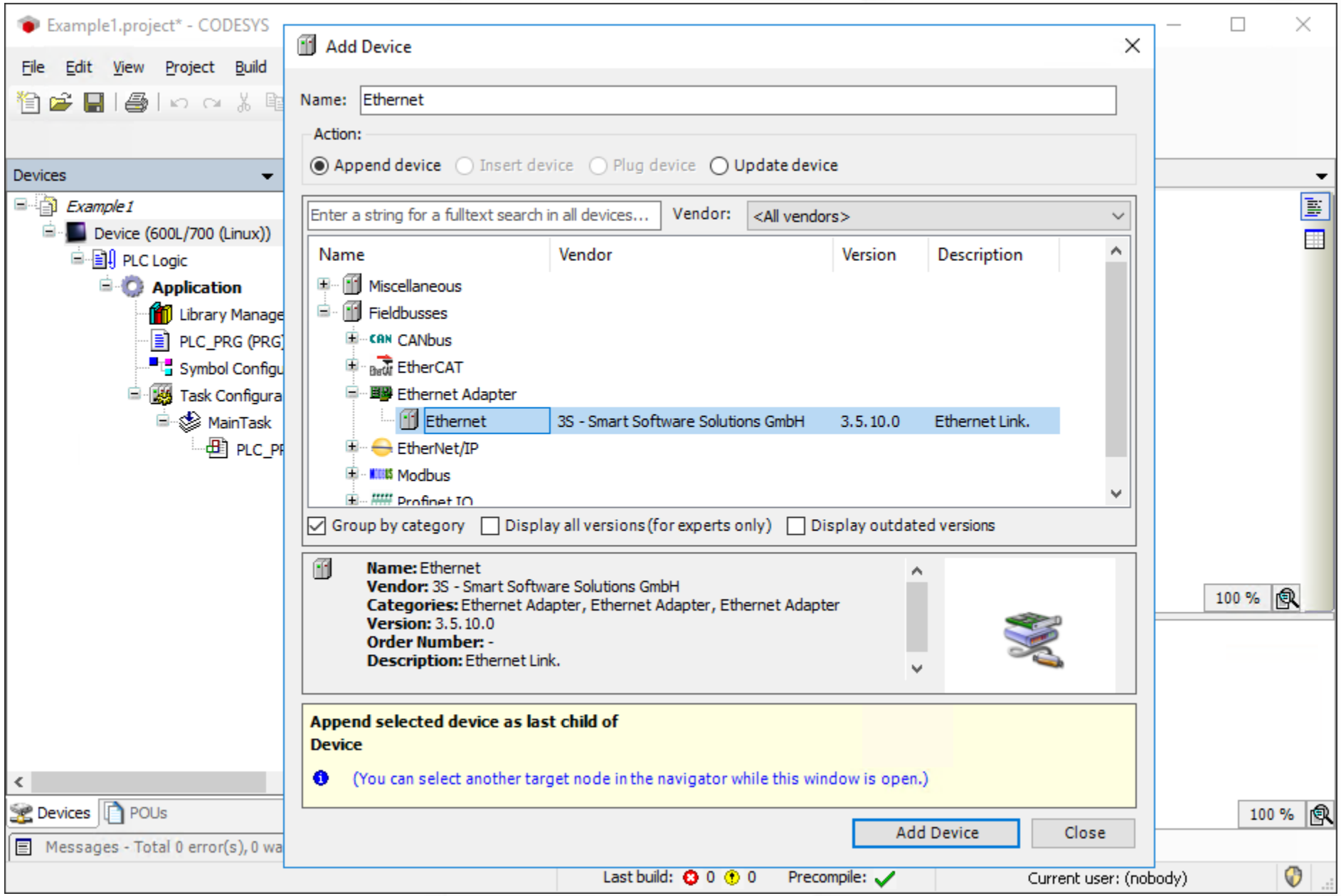
PLC Configuration: add local I/O plug-in



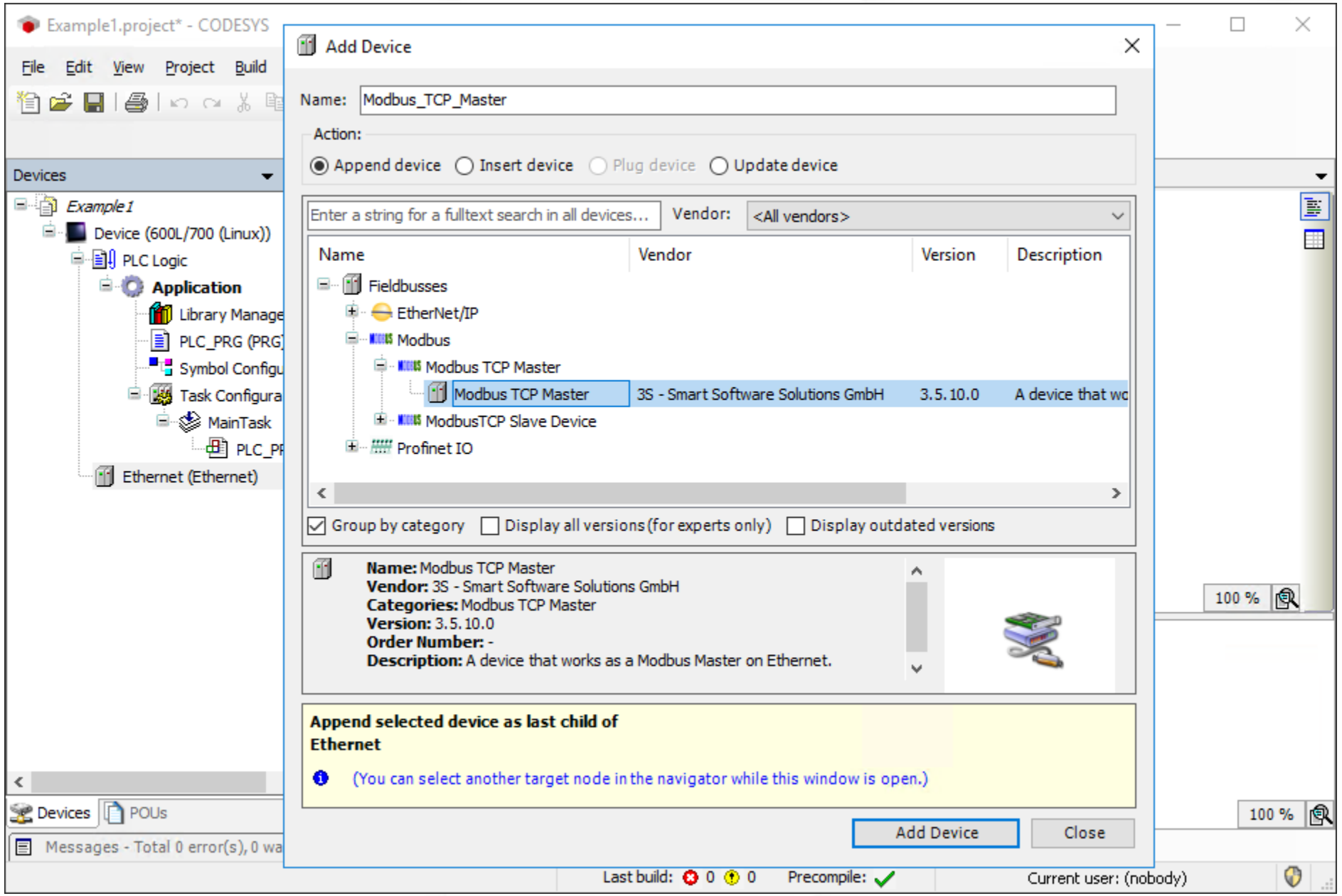
PLC Configuration: add remote I/O



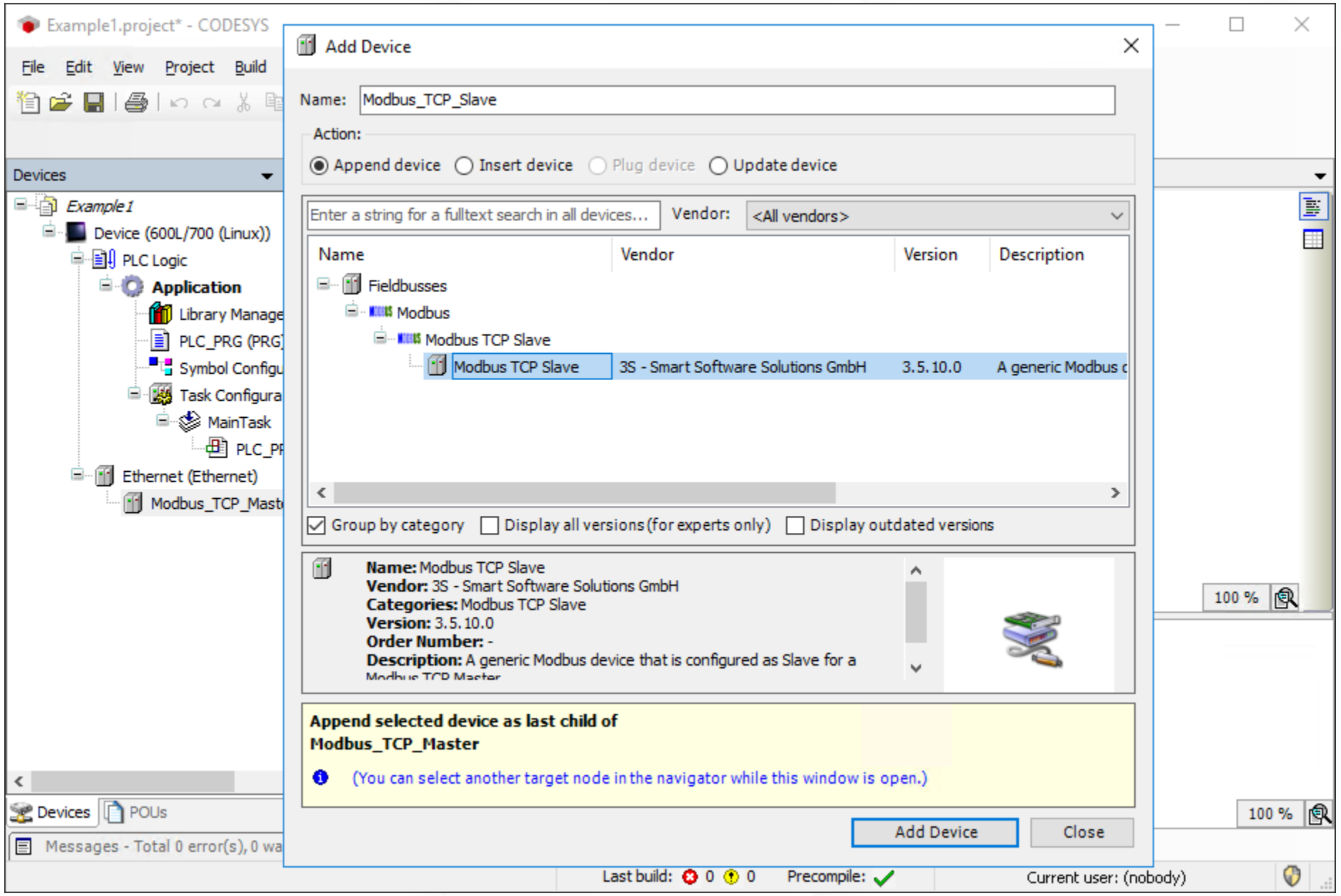
PLC Configuration: add remote I/O



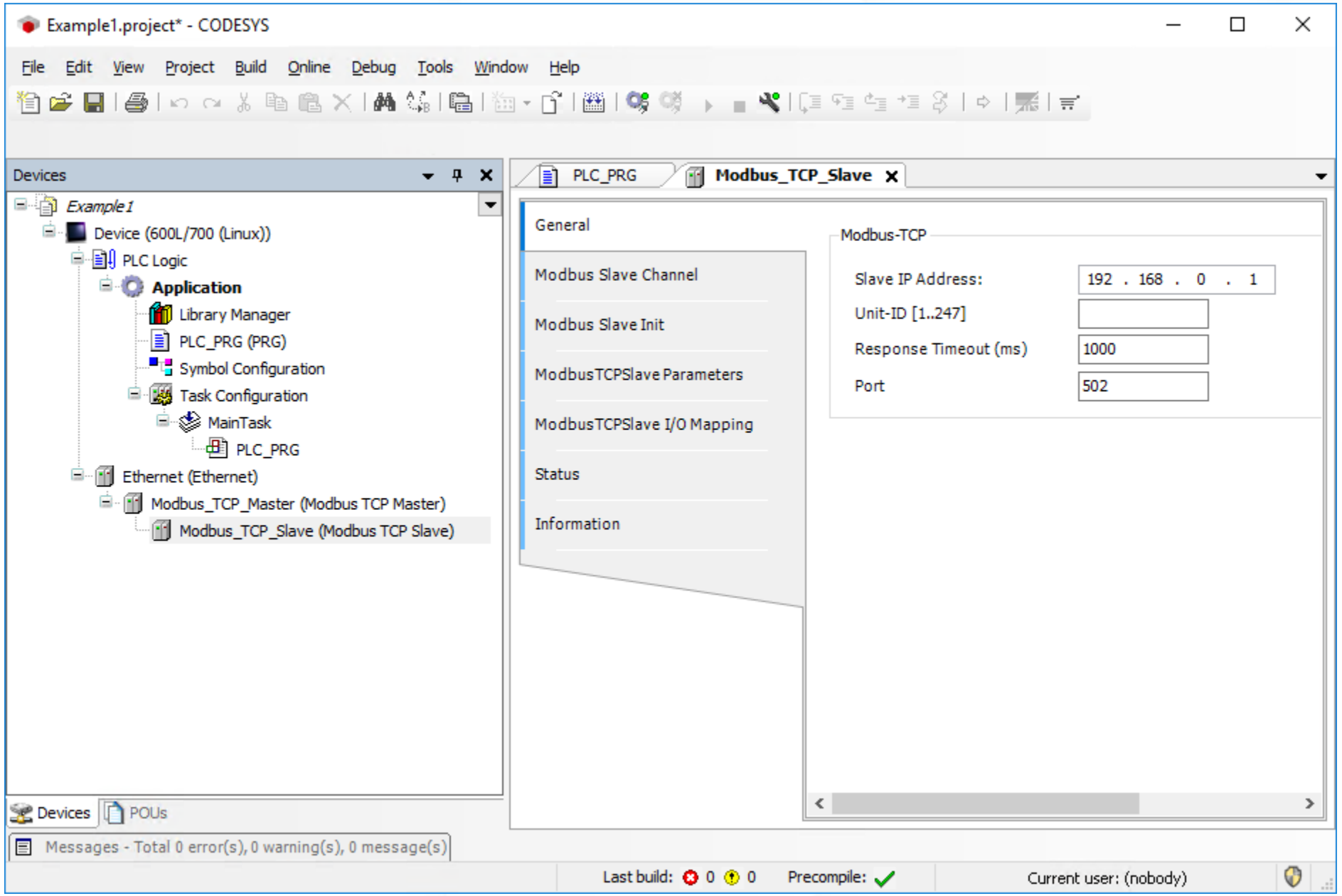
PLC Configuration: add remote I/O



PLC Configuration: add remote I/O



PLC Configuration: add remote I/O



End of JMobile Training Day 2

Thanks for your attention

Technical Support Team