

JMobile Training Day 1

JM 4.0

Agenda

Introduction

Hardware key features

HMI System Settings

Software installation

JMobile Studio IDE

JMobile Features

Introduction

Technical Support

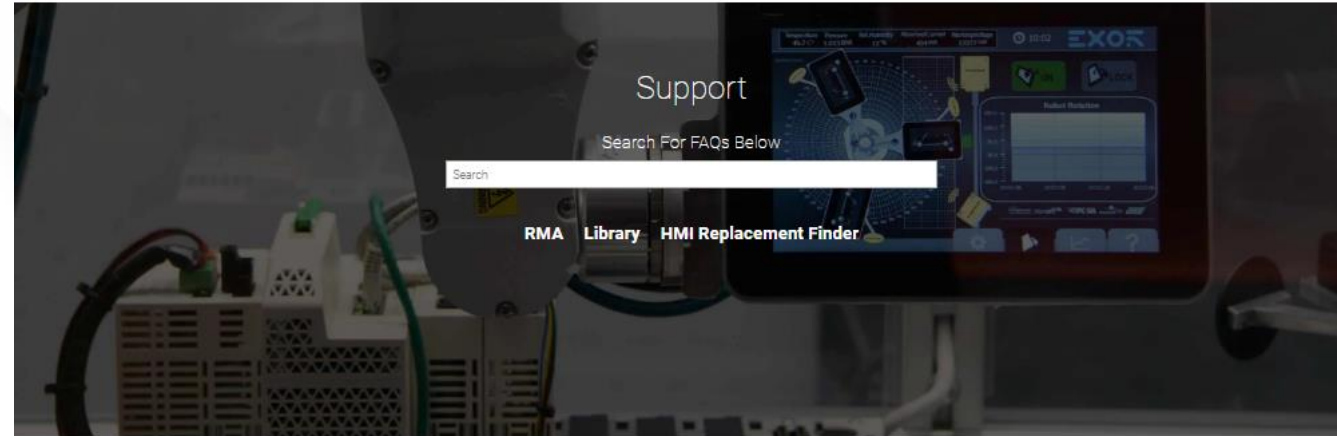
Contact us by:

- Email support.it@exorint.com
- Phone +39 045 8753377
Mon-Fri 9.00-13.00 / 14.00-18.00

Website

<https://www.exorint.com/en/support>

- Frequently Asked Questions
- Demo projects
- Example projects
- Technical notes, cable diagrams



Acquiring Capability

[FAQs](#)

[HMI Replacement Finder](#)

[Support for Suppliers](#)

[JMobile Training](#)

[Webinars](#)

[Corvina Cloud Online Manual](#)

Contact Your Local Support

Italy: support.it@exorint.com

USA: support.us@exorint.com

India: support.in@exorint.com

Germany: support.de@exorint.com

Worldwide: support.it@exorint.com

Maximizing Ownership

[JMobile](#)

[Documents and Download](#)

[UniOP Designer](#)

[Cables](#)

[Technical notes](#)

[Designer Updates](#)

[Repair Service](#)

[RMA](#)

[Documentation](#)

[Library](#)

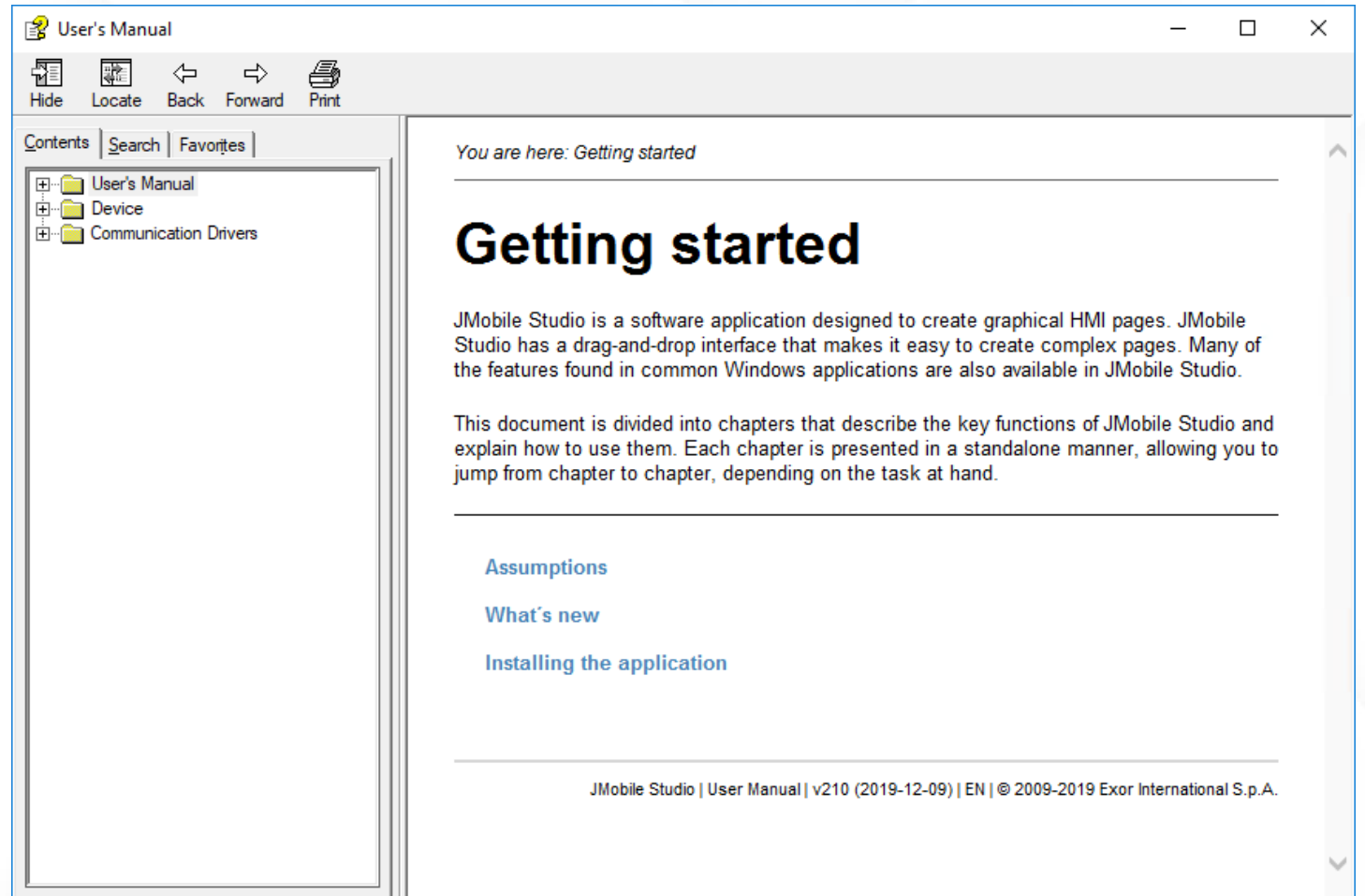
Training and Help

On-line Help includes all topic of this training

- User's manual
- Devices manual
- Communication Drivers

Explanation of features

Technical note with information to setup communication and export PLC variables



What is JMobile?

JMobile is an HMI Software Platform

JMobile is a Suite of software components that provides a complete solution for connecting equipment and visualizing data with EXOR HMIs.

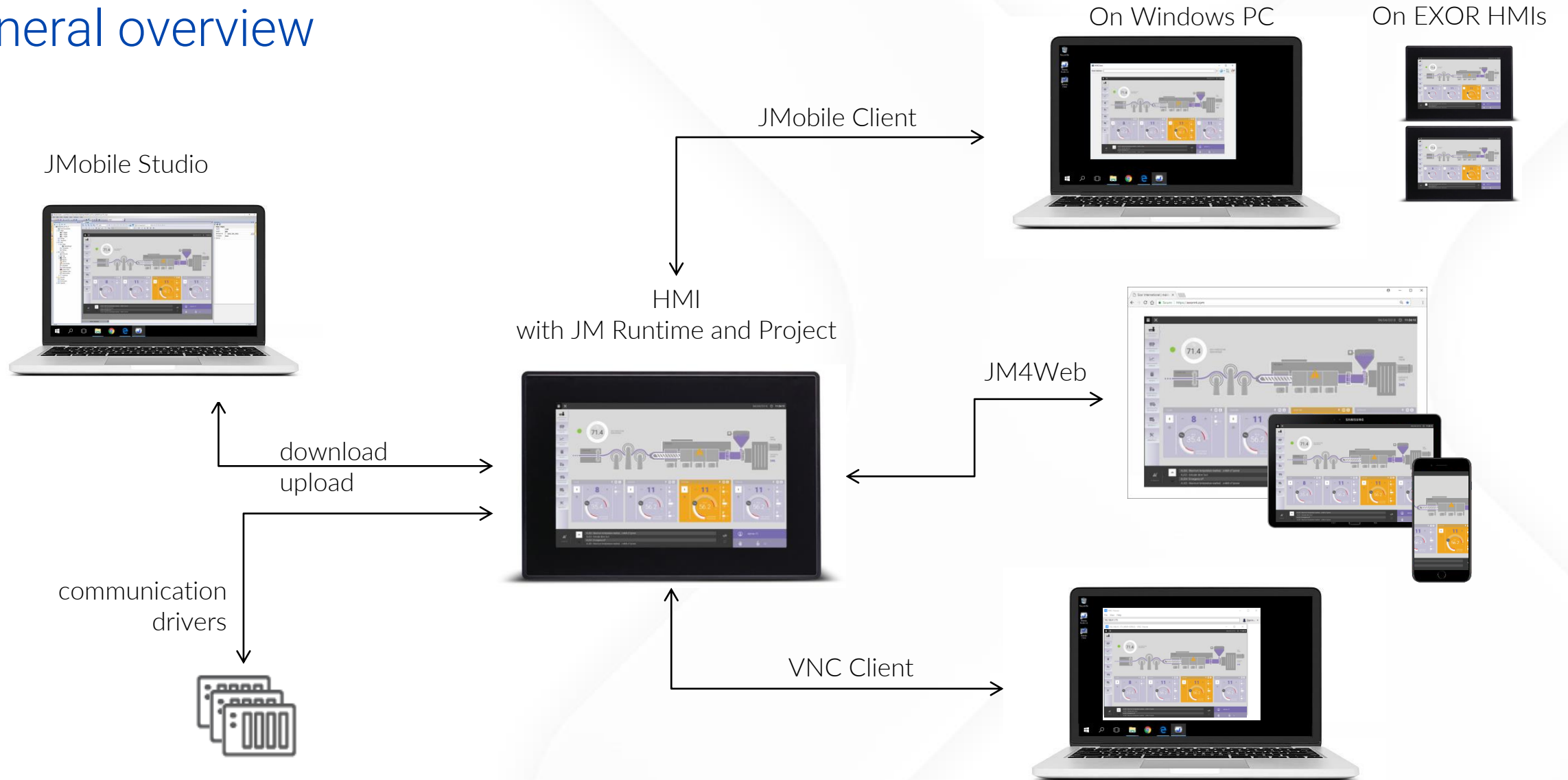


JMobile Suite software package includes:

- JMobile Studio: programming software
- JMobile Runtime PC: HMI engine for industrial PC
- JMobile Client: software for remote control

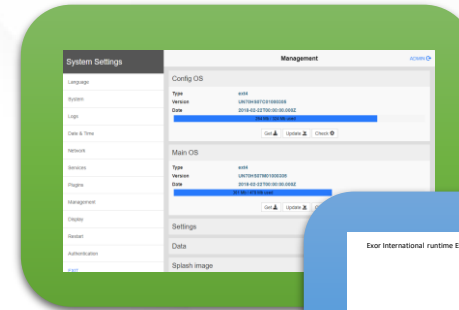


General overview

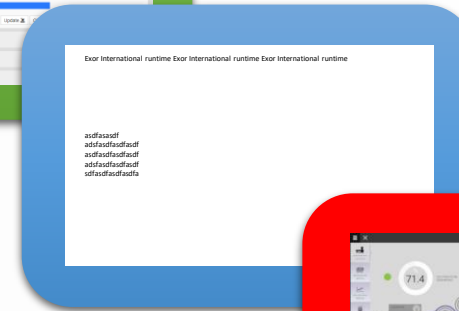


Architecture

BSP – Operating System
Device settings as IP address,
display brightness, services



JMobile Runtime
Software engine capable to run projects



JMobile project
User application created with JMobile Studio



Hardware
key features

JMobile: one software for all devices

All EXOR HMIs and IPCs can be programmed using JMobile Studio

Complete datasheets for hardware and plug in modules are available on website

<https://www.exorint.com/en/products>





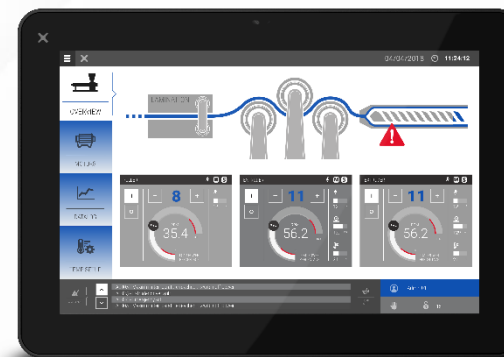
JSmart700 Serie

Linux based

From 5.0" to 21.5" TFT color display,
up to a resolution 1920x1080 pixel
with 16M colors, dimmable backlight

PCAP Touchscreen Multitouch
with swiping and zooming

10/100 Ethernet port PoE
single cable for power and ethernet

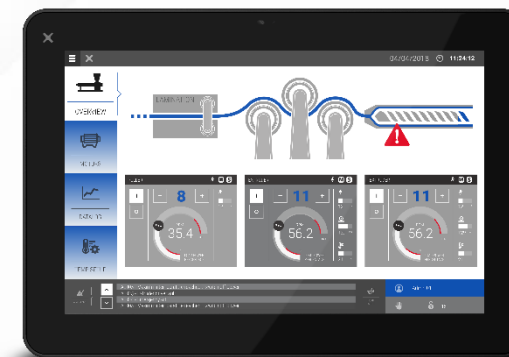


JSmart700 Serie

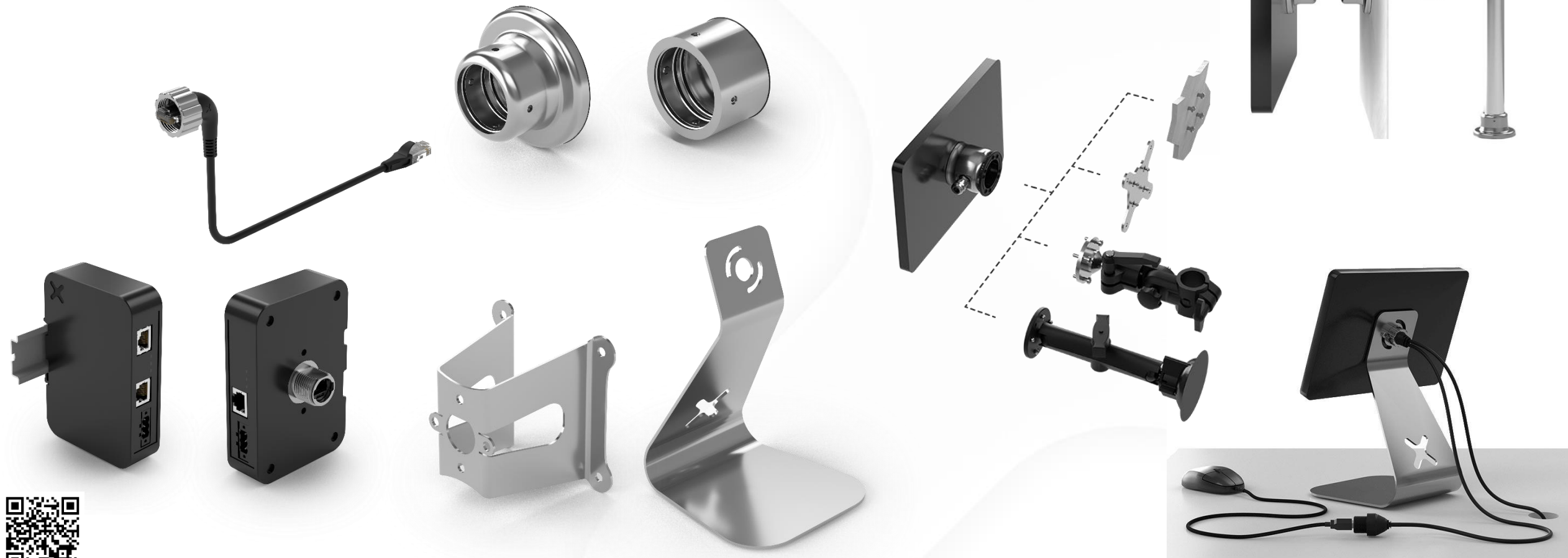
CODESYS V3 with Ethernet I/O Stacks

Wi-Fi Connection

Environment and motion sensors



JSmart700 Serie - Accessories





eX700 Serie

Linux based

From 5.0" to 21.5" TFT color display,
up to a resolution 1920x1080 pixel
with 16M colors, dimmable backlight

PCAP Touchscreen Multitouch
with swiping and zooming

Up to 3 Ethernet ports
Separated networks



eX700 Serie

Plug in Modules for System Expansion
(serial ports, fieldbus, local I/O)

One integrated serial port, two USB ports
One SD Card Slot

CODESYS V3 with Ethernet/Serial/CAN
I/O Stacks





eXware700 Serie

Linux based

Same capabilities of eX700 HMIs

Corvina Cloud VPN license included

IIOT controller





eSMART Serie

Linux based

Powerful entry level HMI

Resistive touch display

CODESYS V3 with Ethernet/Serial I/O Stacks

* no PERSISTENT/RETAIN vars

One serial port

One Ethernet port



eSMART Serie – eSMART107

All eSMART features,
more Hardware capabilities:

Two Ethernet ports

Two USB ports

Two Serial ports (RS-232)

Two Serial ports (RS-485/422 isolated)

Two CAN interfaces





eTOP500 Serie

Windows CE based

Resistive touch display

Two Ethernet ports, one Ethernet adapter

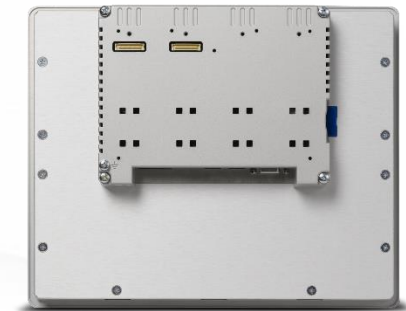


eTOP500 Serie

Plug in Modules for System Expansion
(serial ports, fieldbus, local I/O)

One integrated serial port, two USB ports
One SD Card Slot

CODESYS V3 with Ethernet/Serial/CAN
I/O Stacks



Application oriented



Handhelds



Cabled or
Wireless (Wifi & Bluetooth)



Food and Beverage



IP69
Stainless steel
Chemical resistant



Harsh Environment



High brightness
Long term reliability
Outdoor applications

Industrial PC



Entry-level Panel PC
EPC



High-end Panel PC
IPC



Compact Box Computers
ECC-ECF



HMI System Settings

Linux OS

All new generations HMIs are based on Linux OS

This chapter explains first steps to operate on HMI based on Linux
JSmart700, eX700, eSMART, eXware700

Full detailed documentation is available in JMobile Studio User Manual

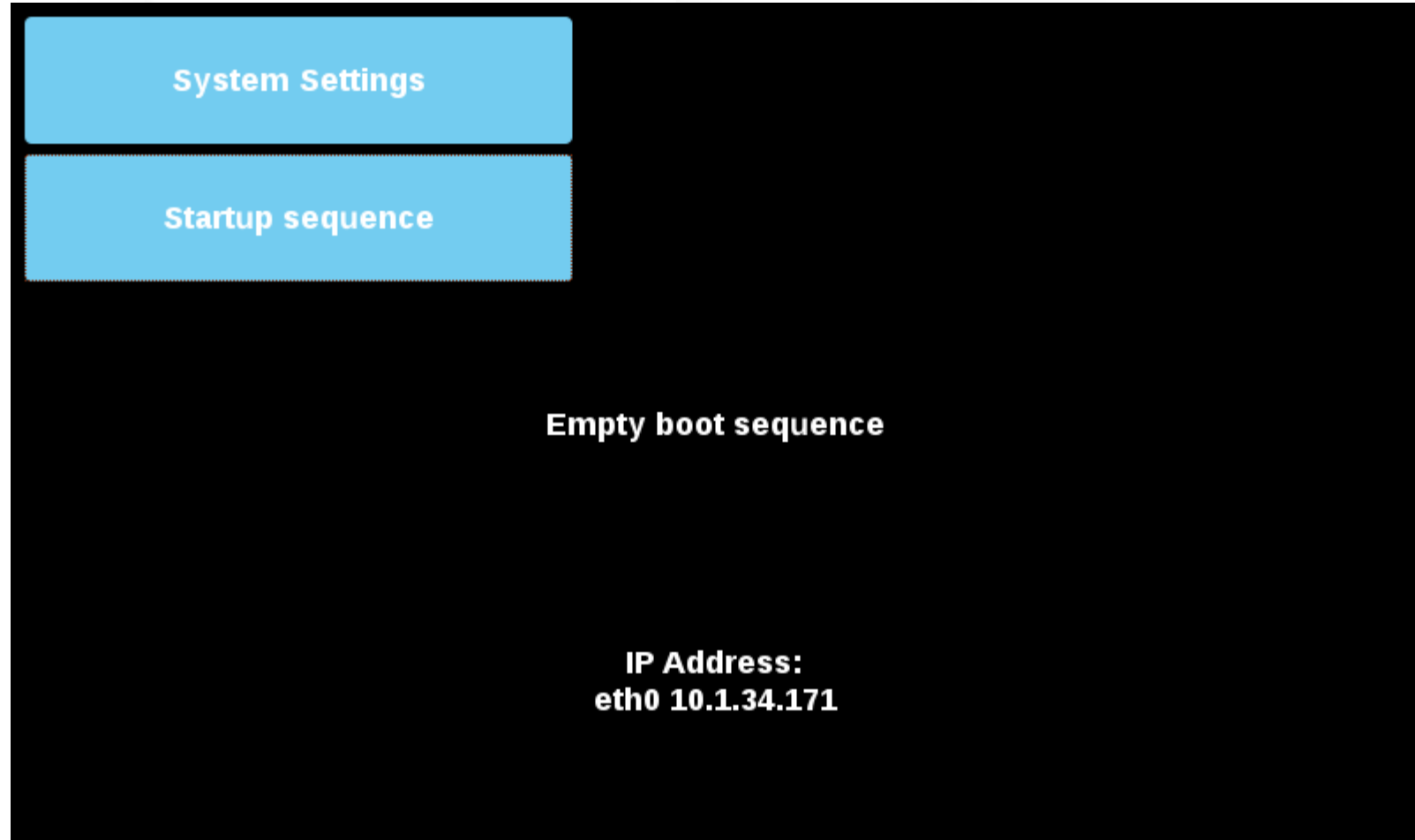
Out of the box

HMIs are delivered without Runtime

Once powered up, shows the "Runtime Loader" screen

Auto IP feature: if no DHCP server is available, HMI auto assign an IP address into the range 169.254.x.x

Runtime can be installed automatically, via Ethernet, on first project download with JMobile Studio manually via USB Memory, creating an "Update Package"



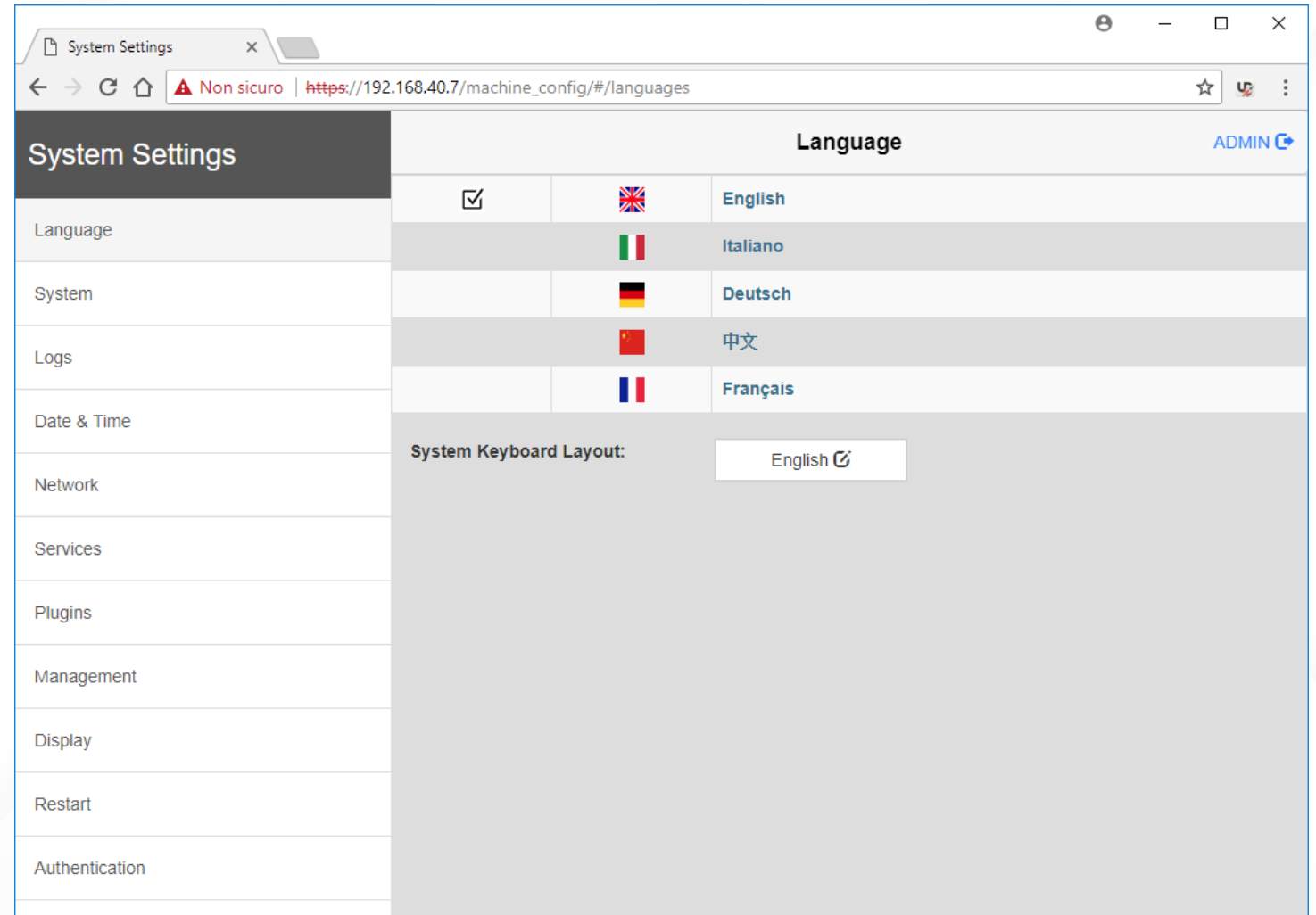
Main OS

System Settings in User Mode (Main OS)

Accessible from "System Settings" item on Runtime Loader or from Contextual Menu

Basic system settings

Can be accessed remotely with browser using URL
`https://<HMI_ip>/machine_config`
`https://<HMI_ip>/system_settings`

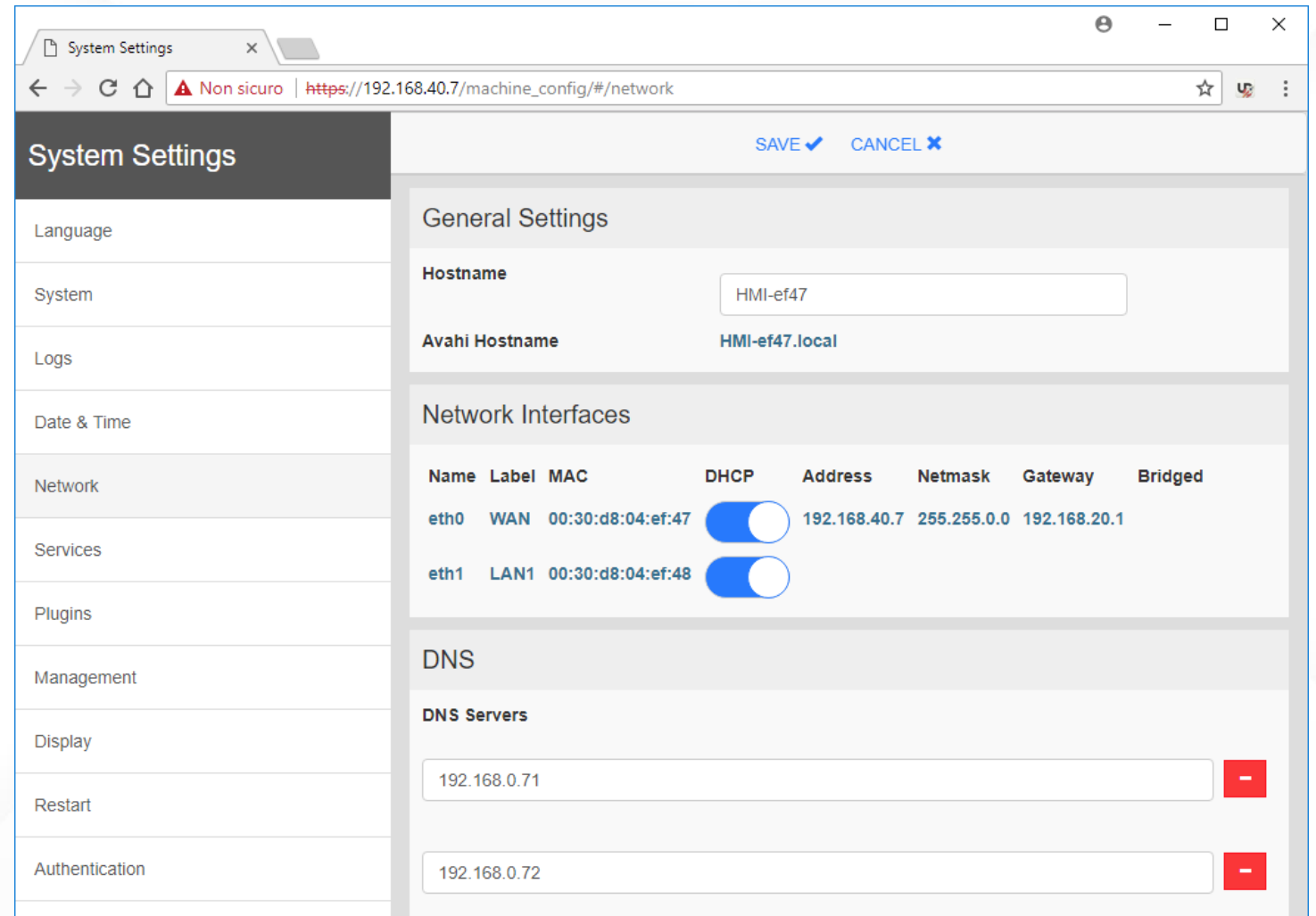


Mainly used settings

Network

Set static or dynamic IP address for any Ethernet interface

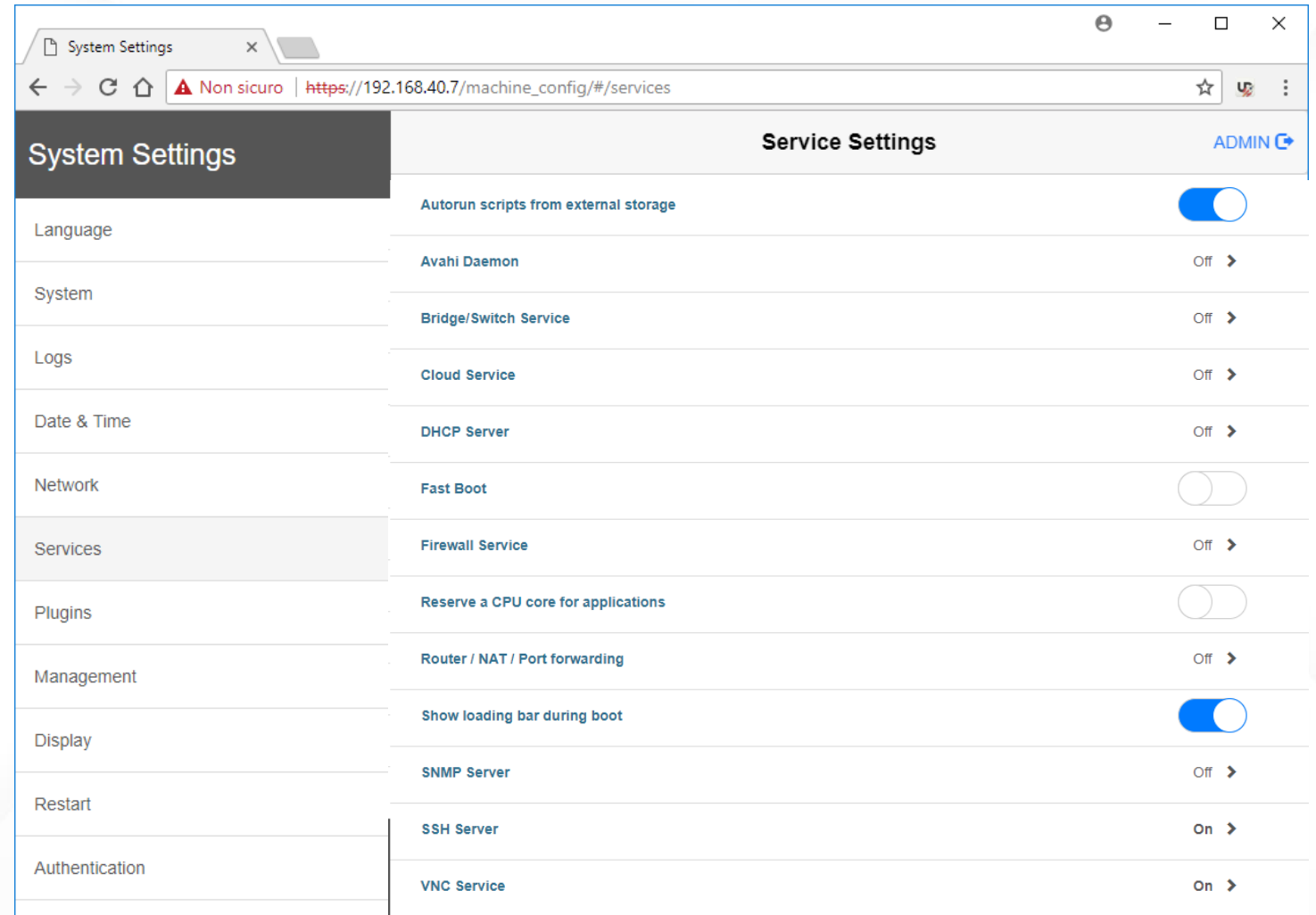
Set DNS to reach the web



Mainly used settings

Services

- VNC remote desktop
- Cloud for VPN control
- SSH to access to Linux bash
- Bridge to connect more ETH
- Firewall, NAT, Routing and Port Forwarding rules
- DHCP Server

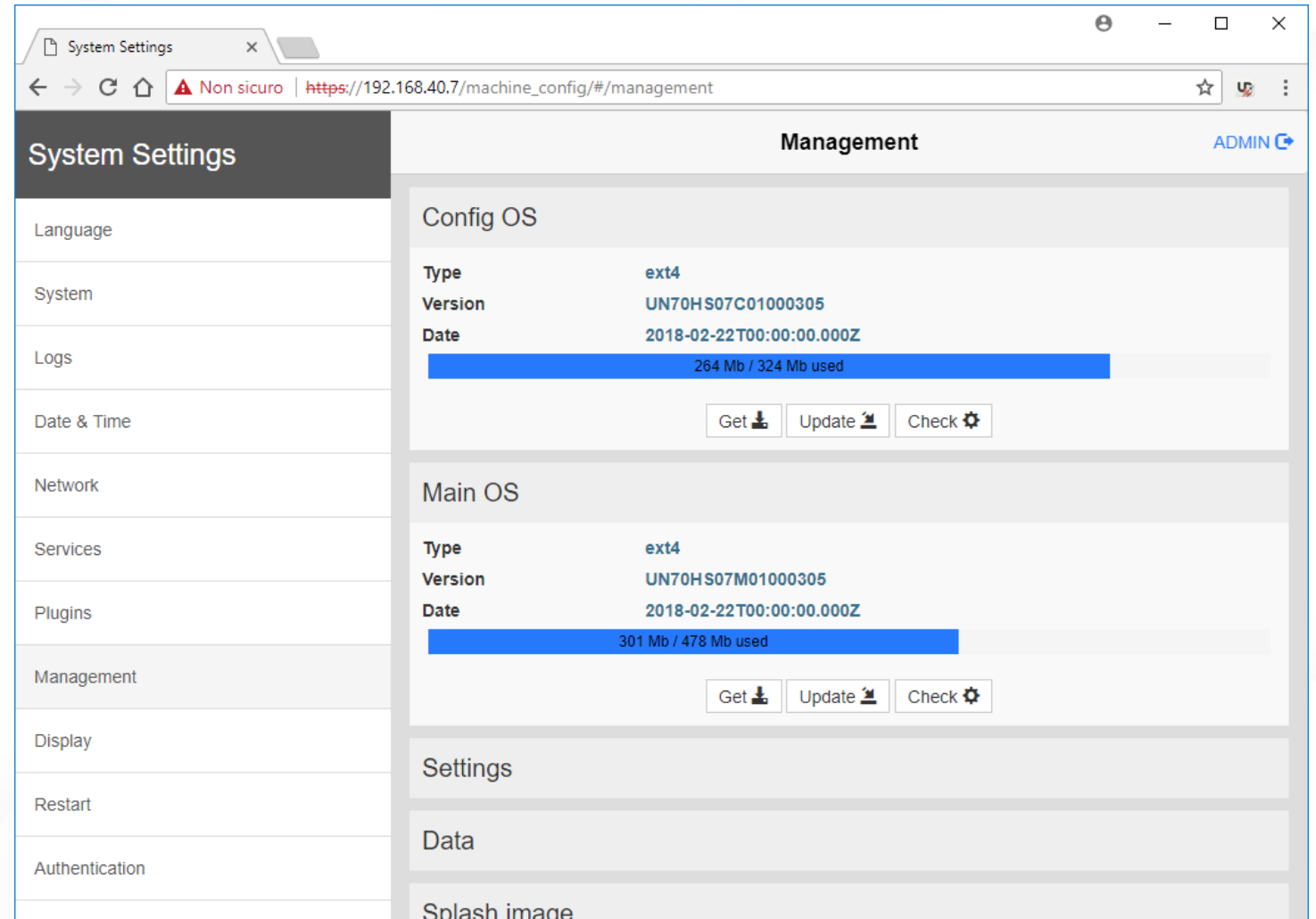


Mainly used settings

Management

Get BSP version

Clear Data and Settings for Factory Reset



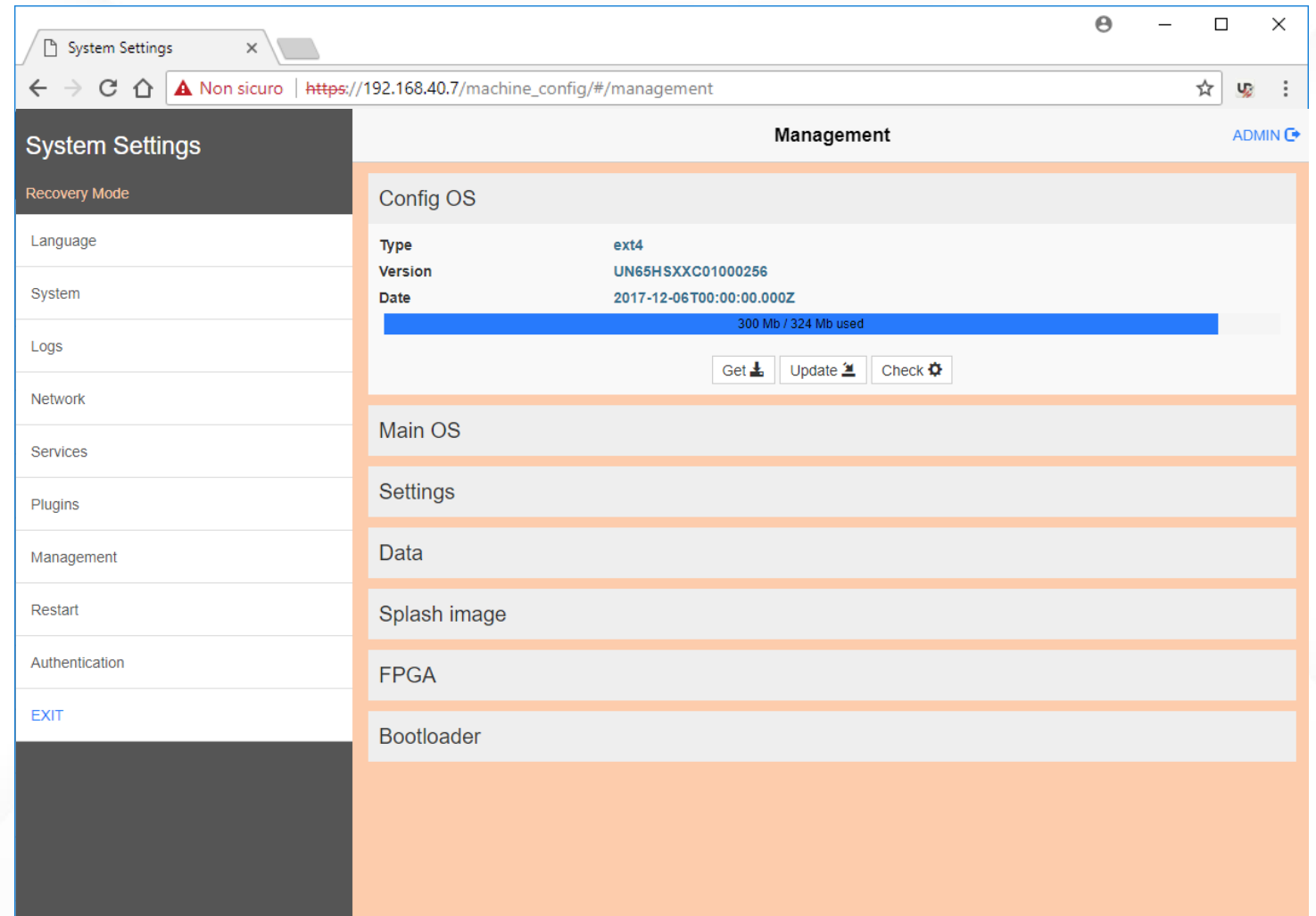
Config OS

System Settings in System Mode (“Tap-Tap” / Config OS):

Accessible by “tap-tap” procedure at power up or from Restart item in User Mode (Config OS)

BSP update

Update can be done remotely with browser using URL
https://<HMI_ip>/machine_config
https://<HMI_ip>/system_settings



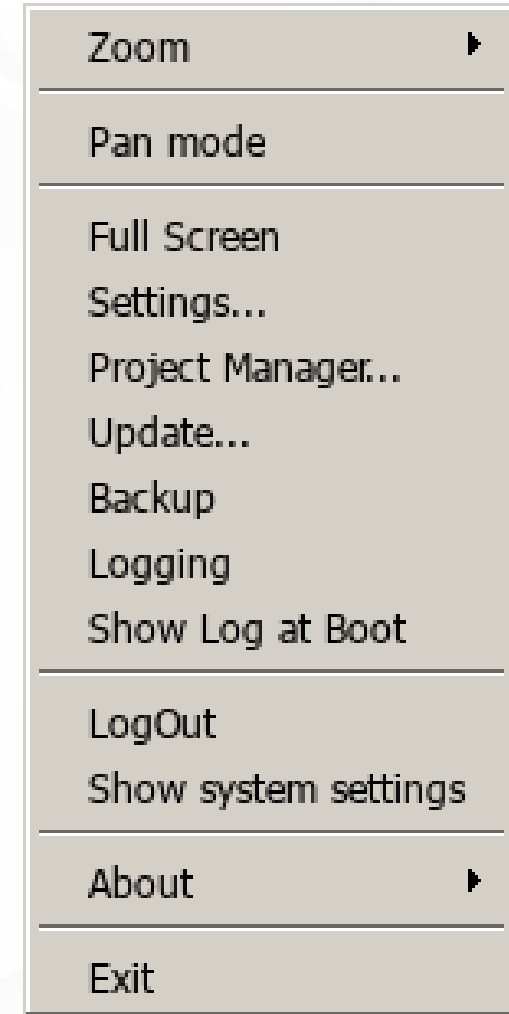
Context Menu

Available when JM Runtime is installed on HMI

Hold on the finger on touch screen for two seconds

Delay can be programmed

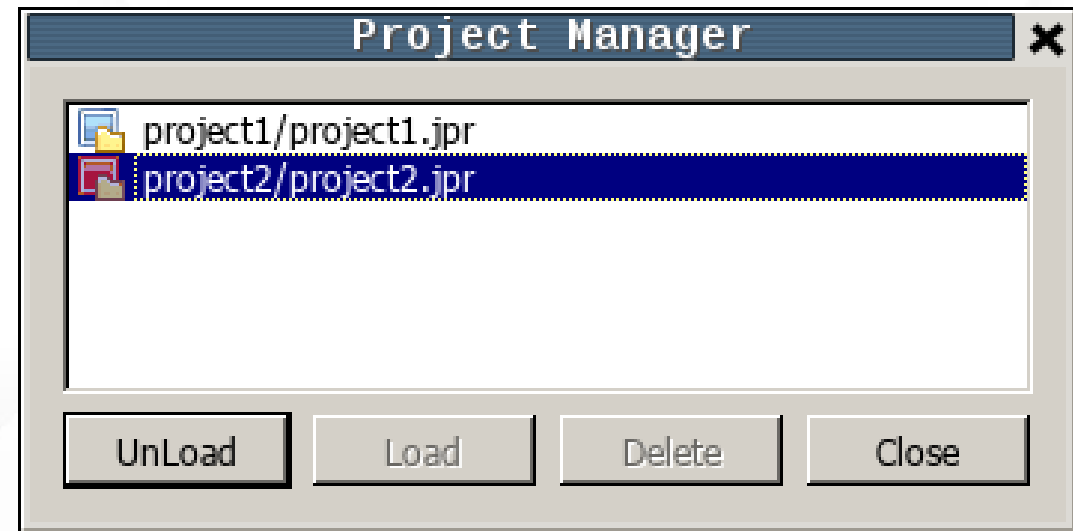
Menu can be "secured" into JMobile Project



Context Menu

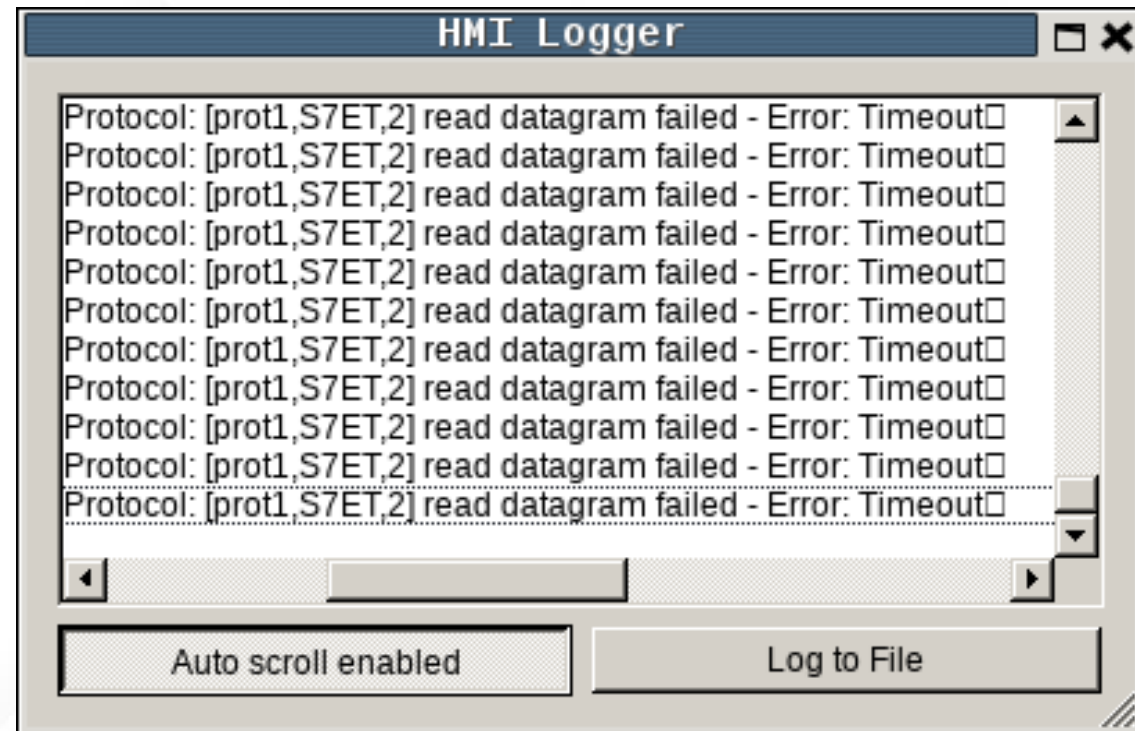
Project Manager to choose which project to load

project can be loaded also from
specific action



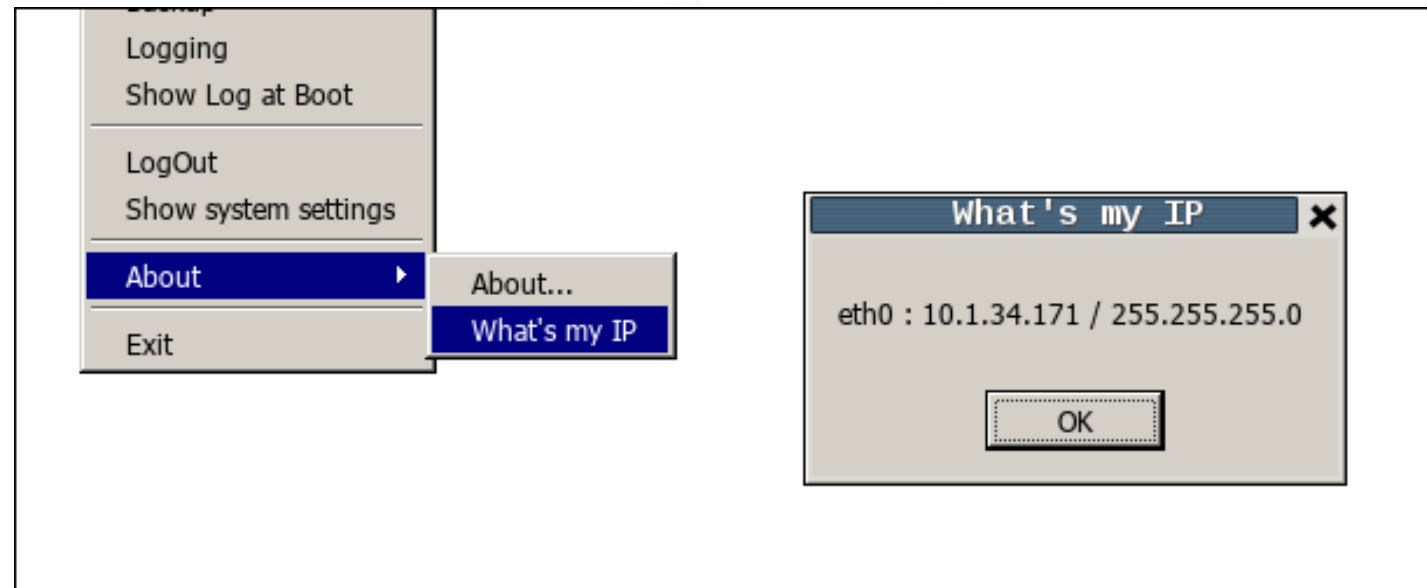
Context Menu

Log info about communication



Context Menu

Quick access to read network settings



Software installation

Installation

JMobile Suite can be downloaded from

<https://www.exorint.com/jmobile-suite-download>

Compile and submit the form to get JMobile Suite download link via email

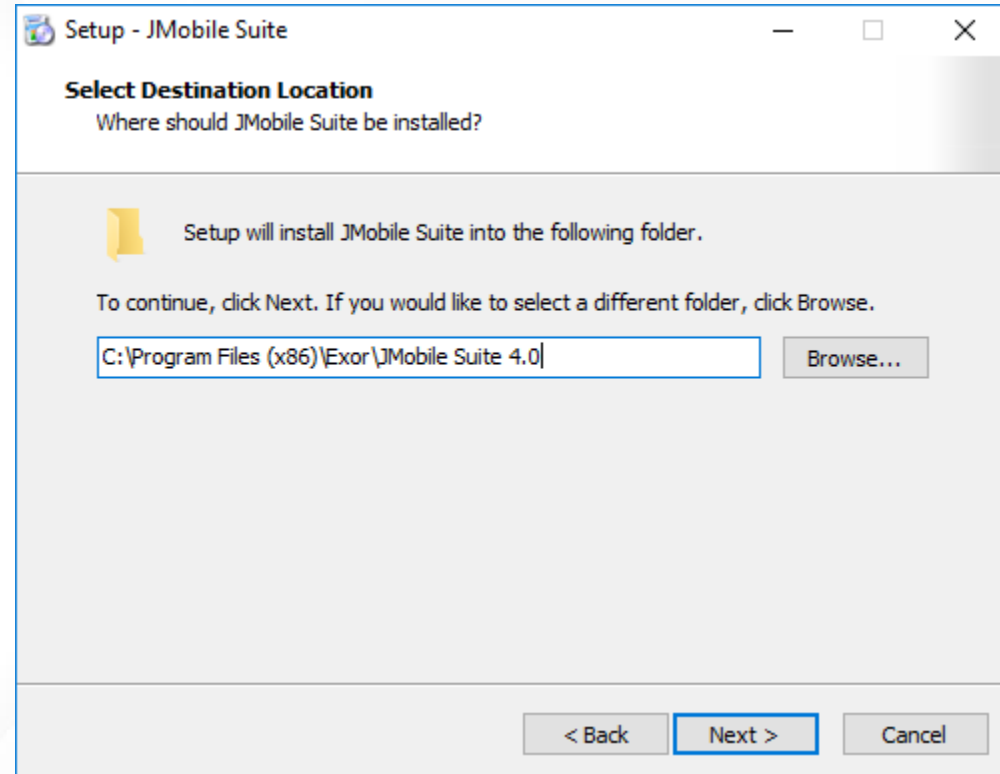
The screenshot shows a web form for downloading JMobile Suite. On the left, the JMobile logo is displayed above the text 'JMobile Suite download form'. Below this, it specifies the version 'v2.8.1.314 - Last update: July 10th 2019' and includes a note: 'Please enter a valid email address. After completing this form, you will receive an email with a download link to the requested file.' On the right side of the form, there are several input fields: 'Email', 'Last name', and 'Company name', each with a corresponding text box. Below these are two dropdown menus for 'Your Company Type' and 'Country/Region', both currently showing 'Please Select'. A 'GDPR' section follows, containing a paragraph of consent text and a checked checkbox labeled 'GDPR Consense'. At the bottom right, there is a blue 'SUBMIT' button.



Installation

Select installation folder

More versions of JMobile can be installed on the same PC

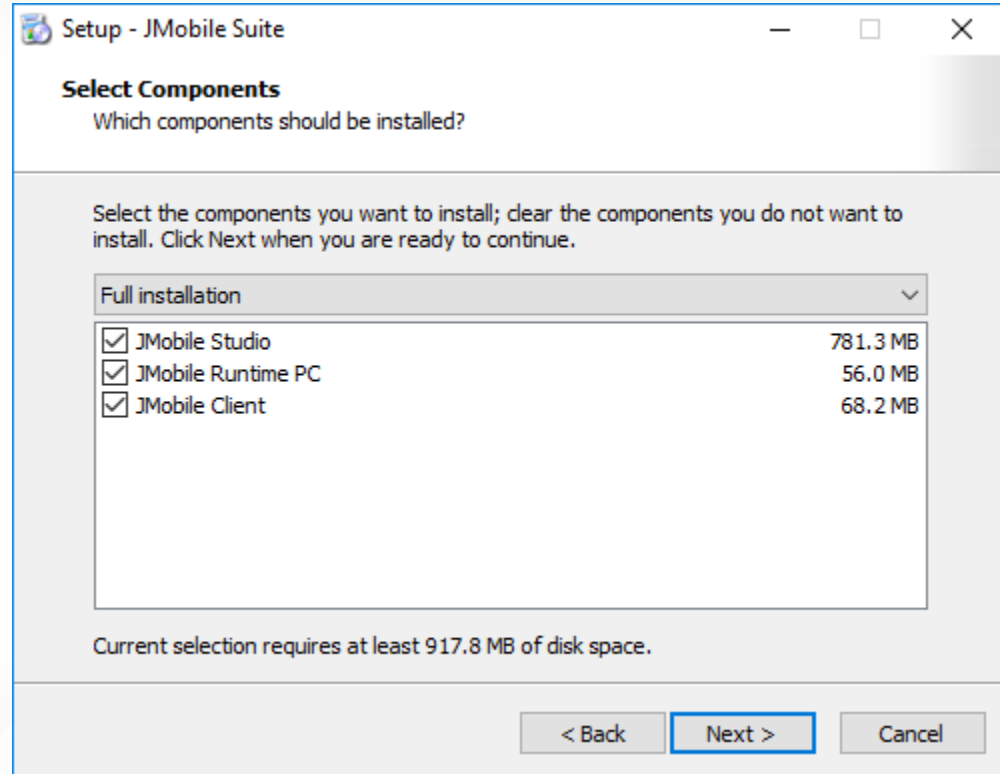


Installation

Select component to be installed

JMobile Studio and
JMobile Runtime PC
need license activation

JMobile Client is free to use



License model

JMobile Studio and JMobile Runtime PC have 30-days free trial fully functional, trial period is not available on Virtual Machine environment.

JMobile Studio license

- License doesn't expire, valid for future upgrades
- One key is valid on a single computer
- Allow execution on JMobile Studio on VM

JMobile Runtime PC license

- License doesn't expire, valid for future upgrades
- One activation key is valid on a single computer
- Execution on VM not allowed even if licensed

License activation

Activation is done via JMobile Studio
Help > Register

Software must be registered filling an
information form

An active Internet connection is needed

If Internet connection is not available
JMobile Studio license can be requested by
email

HMIStudio

First Name*

Last Name*

Job Title

Company*

Division or Group

Industrial Sector

Country*

State/Province*

City*

Phone

E-Mail*

Confirm E-Mail*

[Terms of Service*](#) Yes No *Required Fields

[Privacy Statement*](#) Yes No

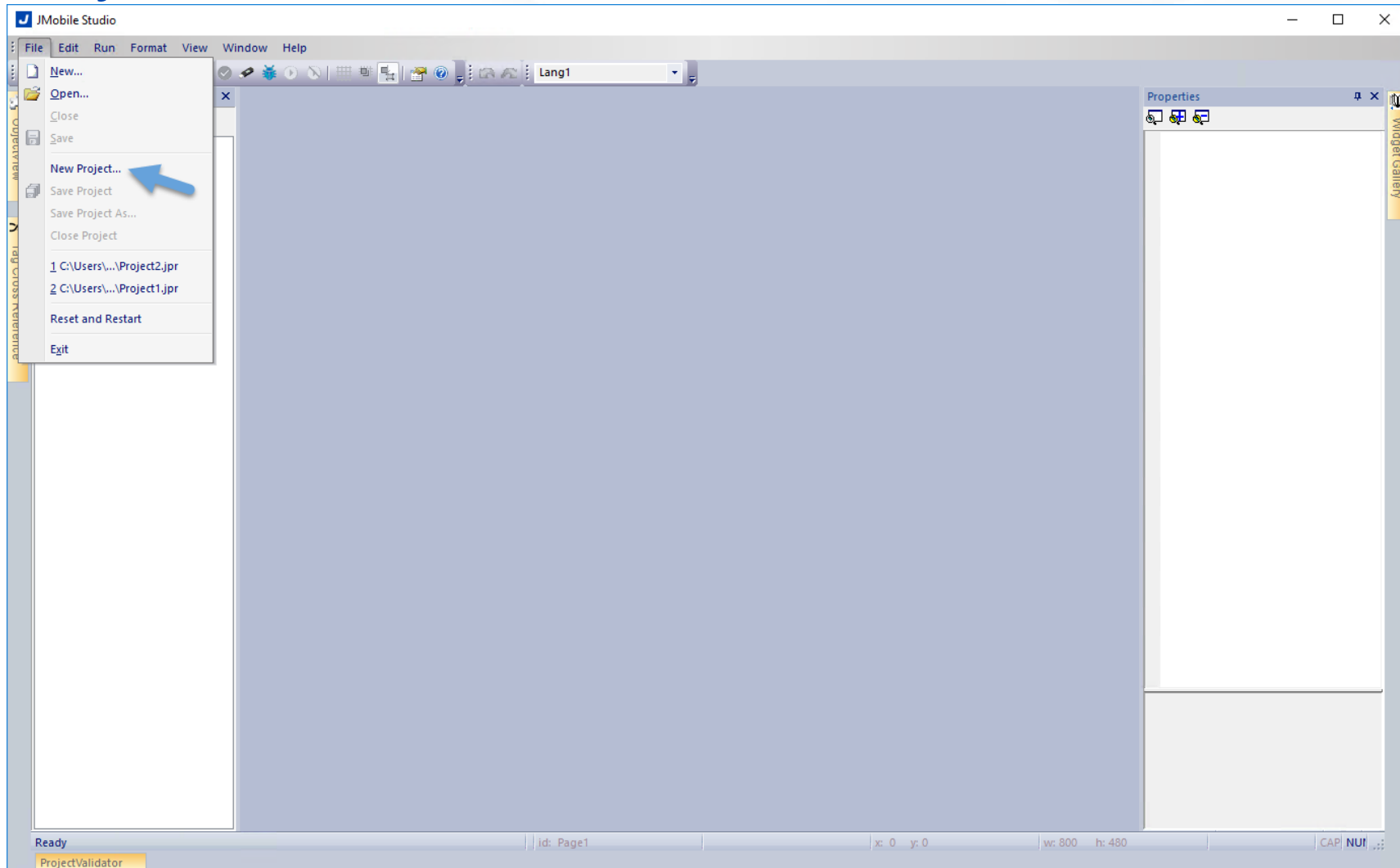
Activation Key*

HostID

JMobile Studio IDE

(Integrated Development Environment)

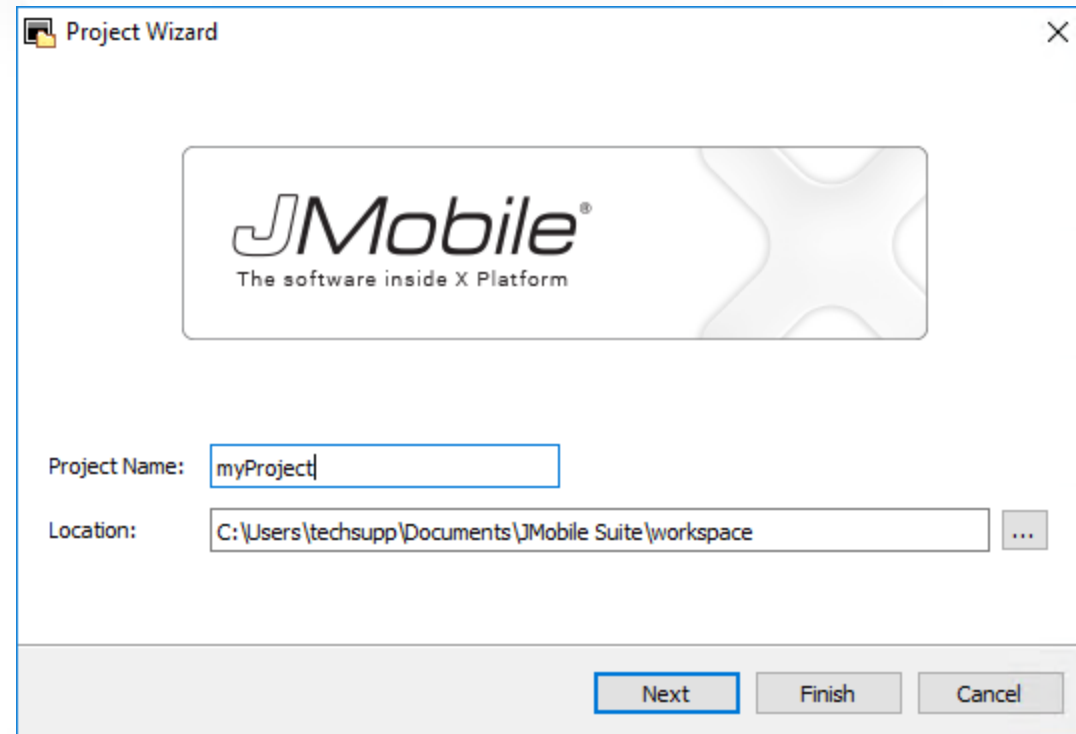
Create New Project



Project Wizard

Wizard for new project

Project Name and Location



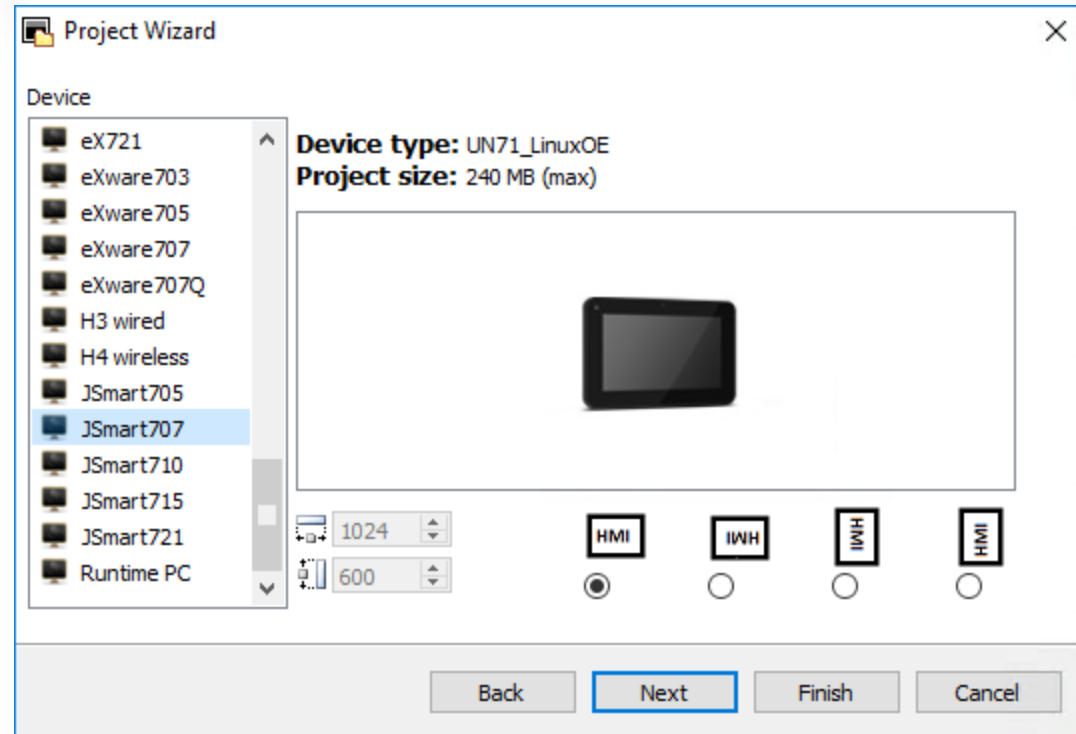
Project Wizard

Target device selection

Max project size

 30MB / 60MB

 60MB / 240MB



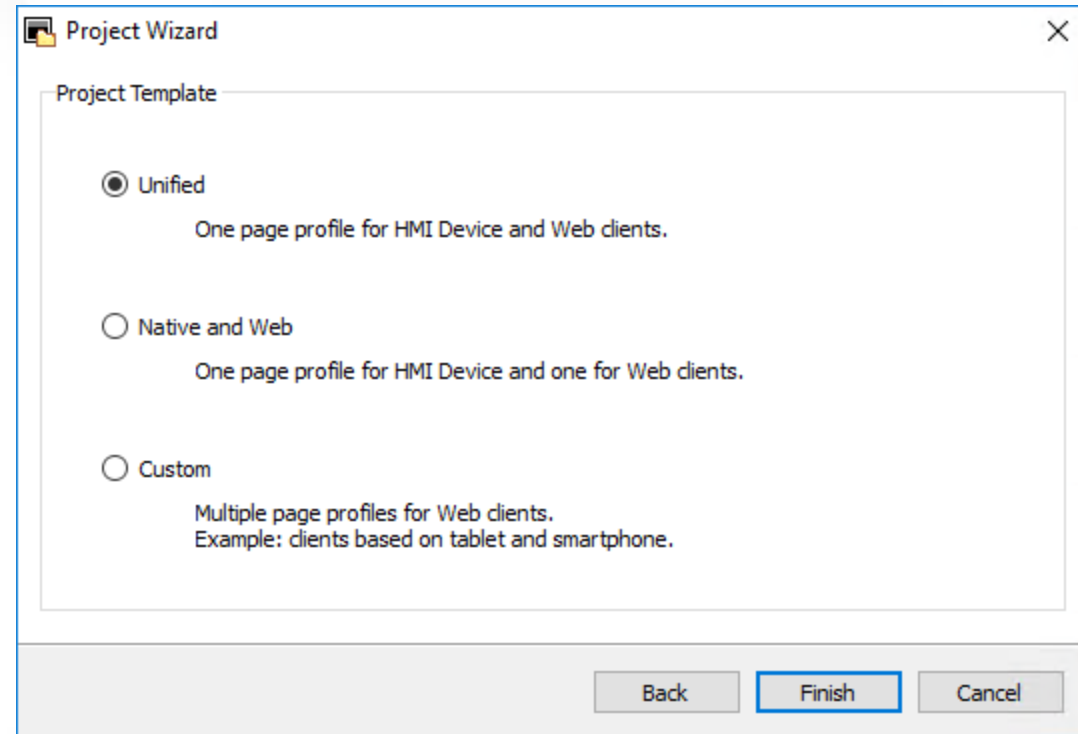
Project Wizard

Page technology selection

Compatible with the past

Unified pages (native and web)

Customized profiles



JMobile Studio Panes

The screenshot displays the JMobile Studio application window with the following panes:

- Objectview (left, green border):** A tree view showing the project structure for 'myProject'. It includes sections for Pages (Unified, Normal), Configuration (Protocols, Tags, Indexed Tag Set, Trends, Reports, Alarms, Events Buffer, Scheduler, MultiLanguage, Screen Saver, Database Links, Data transfers), Interfaces (OPC UA, Corvina, MQTT), Security (UserGroups, Users, AuditTrail), Recipe, Dictionaries, and Keypads (Alphabet, Calendar, Numeric).
- Main Design Canvas (center):** A mobile application interface for a 'LAMINATION' process. It features a top status bar with the date '16/07/2018' and time '18:02:08'. Below is a diagram of a lamination line with rollers. Two large control panels are visible: 'PULLER' with a value of '125' and 'RPM 25.6', and 'EXTRUDER' with a value of '235' and 'RPM 59.8'. Both panels include 'LOAD POWER PERCENTAGE' indicators.
- Widget Gallery (right, red border):** A table showing properties for 'Page : Page1'.

Page : Page1	
Width	1024
Height	600
Fit to Screen Size	Fit to Screen
Background	[255, 25 a +]
Template	none
Events	

Properties pane

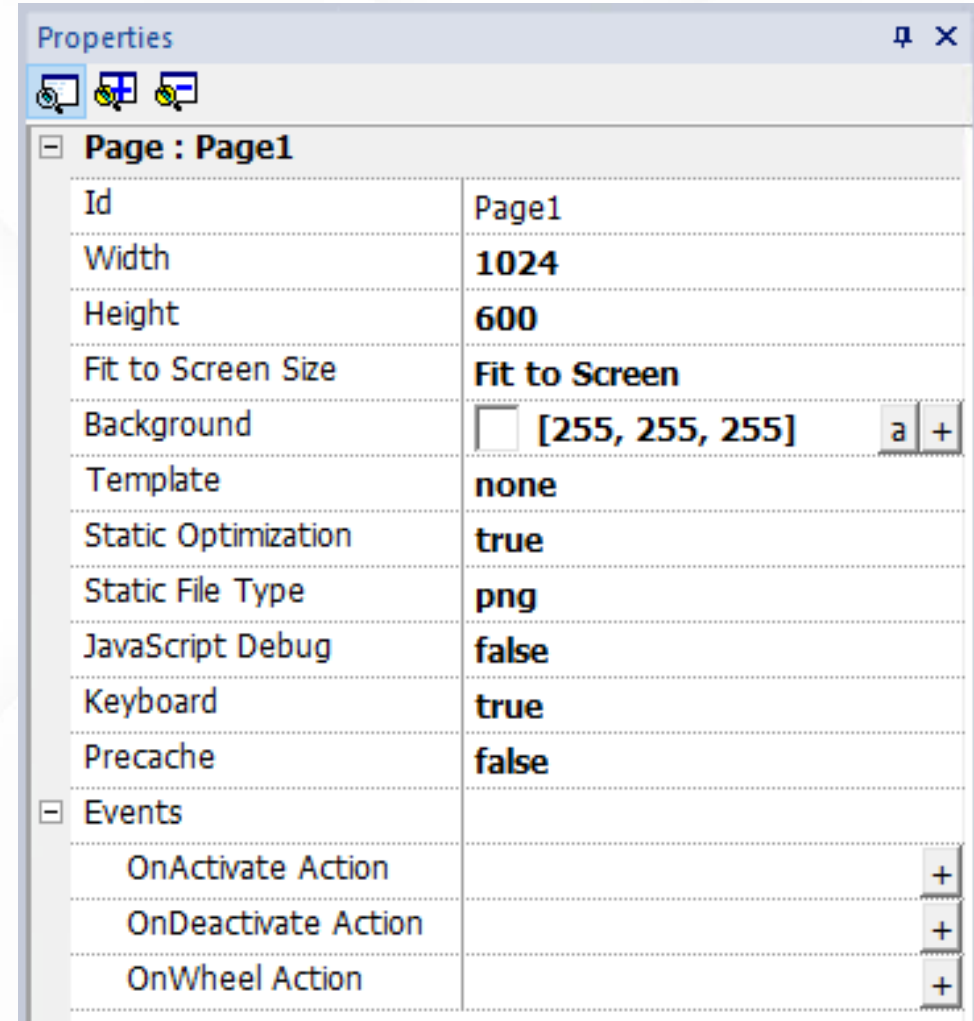
Accessible from pane on the right side

Basic or and Advanced View

All properties that can be attached to tag show [+]
button

Allows to:

- Edit page or widgets properties
- Attach property to a Tag
- Define actions on events



Properties pane: Attach To

Provides the way for attaching Tag to properties

Allows also to show/select Tags from dictionary by "Show all Tags"

field1.value

Source: Tag Alias System Widget Recipe

+ - >> Search Filter by: Data Protocol: Show all Show all tags

Data	Type	Tag name
Modbus TCP:prot1	Container	
Model: Modicon Modbus(1-based)		
Tag1	boolean	Tag1
Tag2	boolean	Tag2
Tag3	boolean	Tag3
Tag4	boolean	Tag4
Tag5	boolean	Tag5
Variables:prot2	Container	
Tag6	short	Tag6
Tag7	short	Tag7
Tag8	short	Tag8
Tag9	short	Tag9
Tag10	short	Tag10

Property	Value
Driver	
Model	Modicon Modbus(1-based)
Protocol	Modbus TCP:prot1
Tag	
Data Type	boolean
Tag name	Tag1
PLC tag name	
Groups	
Tag URI	?INP?100001?boolean
Comment	
Rate	500
R/W	R/W

Ready

Read Only Read/Write Write Only *Items used: 10/10000* Array index: 0

Formula

f- Scaling

Bit/Byte Indexing

Color Palette

OK Cancel

Properties pane: Attach To

Expressions available for transforms associated to Tags

Using JavaScript syntax

Auto Syntax check on saving

Direct tag access: \$("Tag1")

Save & reuse a formula

Set of operators with helper

Math: (+, -, *, /)

Logic: (Not, And, Or)

Compare (<, <=, >, >=, ==)

field1.value

Source: Tag Alias System Widget Recipe

+ - >> Search Filter by: Data Protocol: Show all Show all tags

Data	Type	Tag name
Modbus TCP:prot1	Container	
Model: Modicon Modbus(1-based)		
Tag1	boolean	Tag1
Tag2	boolean	Tag2
Tag3	boolean	Tag3
Tag4	boolean	Tag4
Tag5	boolean	Tag5
Variables:prot2	Container	
Tag6	short	Tag6
Tag7	short	Tag7
Tag8	short	Tag8
Tag9	short	Tag9
Tag10	short	Tag10

Property	Value
Driver	
Model	Modicon Modbus(1-based)
Protocol	Modbus TCP:prot1
Tag	
Data Type	boolean
Tag name	Tag1
PLC tag name	
Groups	
Tag URI	?INP?10000?boolean
Comment	
Rate	500
R/W	R/W

Ready

Read Only Read/Write Write Only *Items used: 10/10000* Array index: 0

Formula

Scaling

Bit/Byte Indexing

Color Palette

Function

- System
 - Abs(argument)
 - And(value1,value2)
 - Bit(argument,bitNumber)
 - Byte(argument,byteNumber)
 - CharAt(targetString,index)
 - Contains(targetString,keyword)
 - ContainsRegExp(targetString,regExp)
 - Cos(argument)
 - Exp(argument)
- User
 - sum(Tag1,Tag2)

OK Cancel

Properties pane: Attach To

Applies scaling on Tag visualization

By Formula

Defining linear scaling through parameters

By Range

Calculate formula automatically defining Input and Output

field1.value

Source: Tag Alias System Widget Recipe

+ - >> Search Filter by: Data Protocol: Show all Show all tags

Data	Type	Tag name
Modbus TCP:prot1	Container	
Model: Modicon Modbus(1-based)		
Tag1	boolean	Tag1
Tag2	boolean	Tag2
Tag3	boolean	Tag3
Tag4	boolean	Tag4
Tag5	boolean	Tag5
Variables:prot2	Container	
Tag6	short	Tag6
Tag7	short	Tag7
Tag8	short	Tag8
Tag9	short	Tag9
Tag10	short	Tag10

Property	Value
Driver	
Model	Modicon Modbus(1-...
Protocol	Modbus TCP:prot1
Tag	
Data Type	boolean
Tag name	Tag1
PLC tag name	
Groups	
Tag URI	?INP?10000?1?boo...
Comment	
Rate	500
R/W	R/W

Ready

Read Only Read/Write Write Only Items used: 10/10000 Array index 0

Formula

f Scaling

By Formula By Range

1.00 x Value + 0.00

Input Output

Min: 0 Min: 0

Bit/Byte Indexing

Color Palette

OK Cancel

Properties pane: Attach To

Points to specific
Byte or Bit

field1.value

Source: Tag Alias System Widget Recipe

+ - >> Search Filter by: Data Protocol: Show all Show all tags

Data	Type	Tag name
Modbus TCP:prot1	Container	
Model: Modicon Modbus(1-based)		
Tag1	boolean	Tag1
Tag2	boolean	Tag2
Tag3	boolean	Tag3
Tag4	boolean	Tag4
Tag5	boolean	Tag5
Variables:prot2	Container	
Tag6	short	Tag6
Tag7	short	Tag7
Tag8	short	Tag8
Tag9	short	Tag9
Tag10	short	Tag10

Property	Value
Driver	
Model	Modicon Modbus(1-...
Protocol	Modbus TCP:prot1
Tag	
Data Type	boolean
Tag name	Tag1
PLC tag name	
Groups	
Tag URI	1?INP?100001?boo...
Comment	
Rate	500
R/W	R/W

Ready

Read Only Read/Write Write Only *Items used: 10/10000* Array index 0

Formula

f Scaling

Bit/Byte Indexing

Byte index 1

Bit index 0

Color Palette

OK Cancel

Properties pane: Attach To

Maps colors to Tag values by

Single Values

5 > Color

Range

0-5 > Color

List

1, 5-10 > Color

10-20, 30-50 > Color

field1.value

Source: Tag Alias System Widget Recipe

Filter by: Data Protocol: Show all Show all tags

Data	Type	Tag name
Modbus TCP:prot1	Container	
Model: Modicon Modbus(1-based)		
Tag1	boolean	Tag1
Tag2	boolean	Tag2
Tag3	boolean	Tag3
Tag4	boolean	Tag4
Tag5	boolean	Tag5
Variables:prot2	Container	
Tag6	short	Tag6
Tag7	short	Tag7
Tag8	short	Tag8
Tag9	short	Tag9
Tag10	short	Tag10

Property	Value
Driver	Modicon Modbus(1-...
Model	Modicon Modbus(1-...
Protocol	Modbus TCP:prot1
Tag	
Data Type	boolean
Tag name	Tag1
PLC tag name	
Groups	
Tag URI	1?INP?100001?boo...
Comment	
Rate	500
R/W	R/W

Ready

Read Only Read/Write Write Only *Items used: 10/10000* Array index 0

Formula

Scaling

Bit/Byte Indexing

Color Palette

Tag Value	Output Color
1 0	#00ff7f
2 1	#ff00ff

OK Cancel

Properties pane: Events and Actions

Fields

Events	
OnDataUpdate Action	+

Buttons

Events	
OnMouseClicked Action	+
OnMouseHold Action	+
OnMousePress Action	+
OnMouseRelease Action	+
OnDataUpdate Action	+

Page

Events	
OnActivate Action	+
OnDeactivate Action	+
OnWheel Action	+

Alarm events

Scheduler events

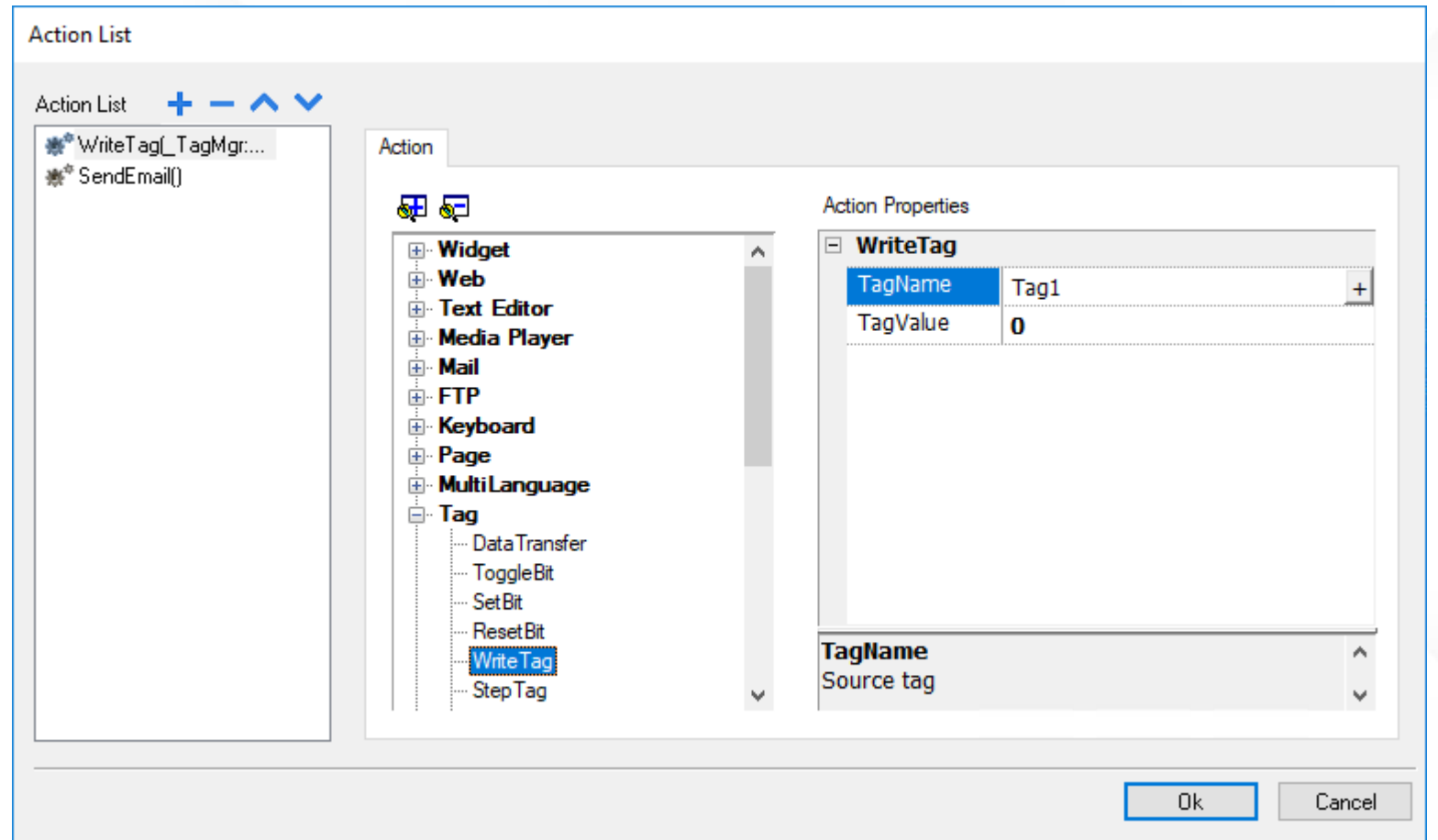
Properties pane: Events and Actions

Actions are executed when an Event is triggered

Support for multiple Actions

List of predefined available Actions

Custom Action with scripting using JavaScript



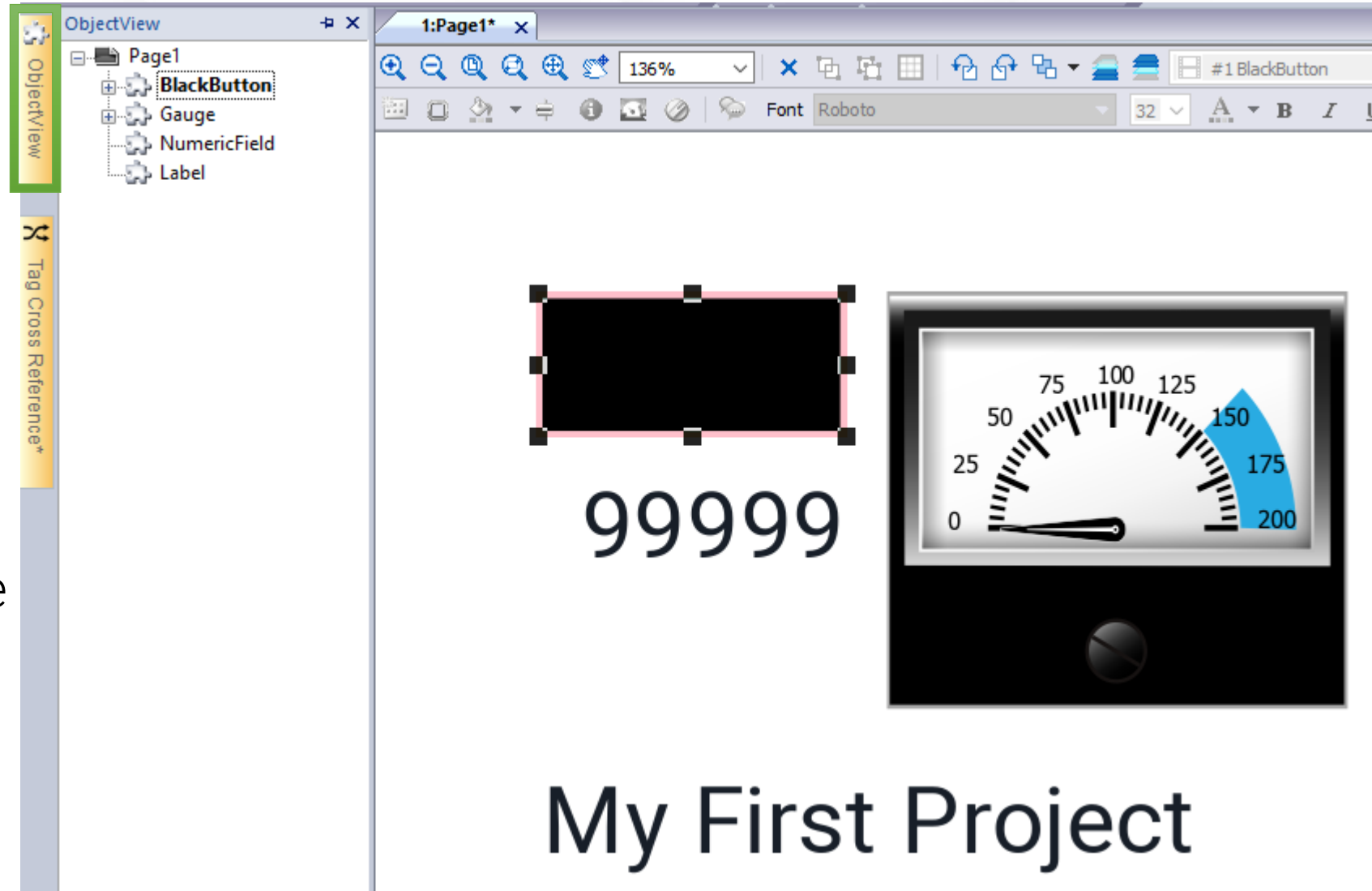
ObjectView pane

Accessible from slide-in pane on the left side

Map of the objects in page

Shows tree view of object composition in page

Locate objects from pane to page and vice-versa



Tag Cross Reference pane

The Tag Cross Reference displays Tags used into project, based on their location

References can be grouped by

Tag: every tag is showed where it's used;

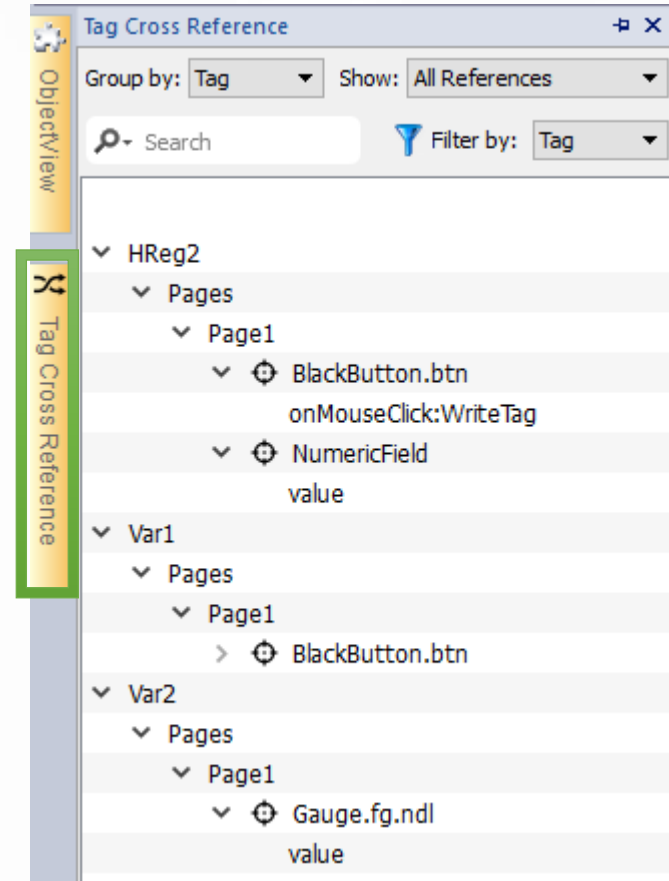
Location: every project location (pages, alarms..) shows which tags are used.

Cross Reference pane shows:

All References

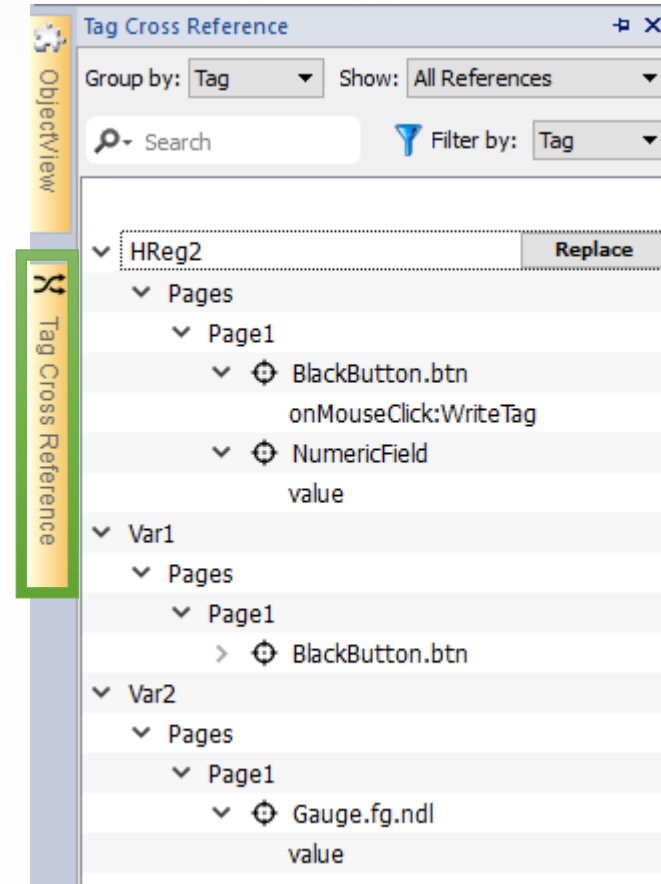
Invalid Tag reference

Unused tags



Tag Cross Reference pane

Possibility to replace Tags in all occurrences

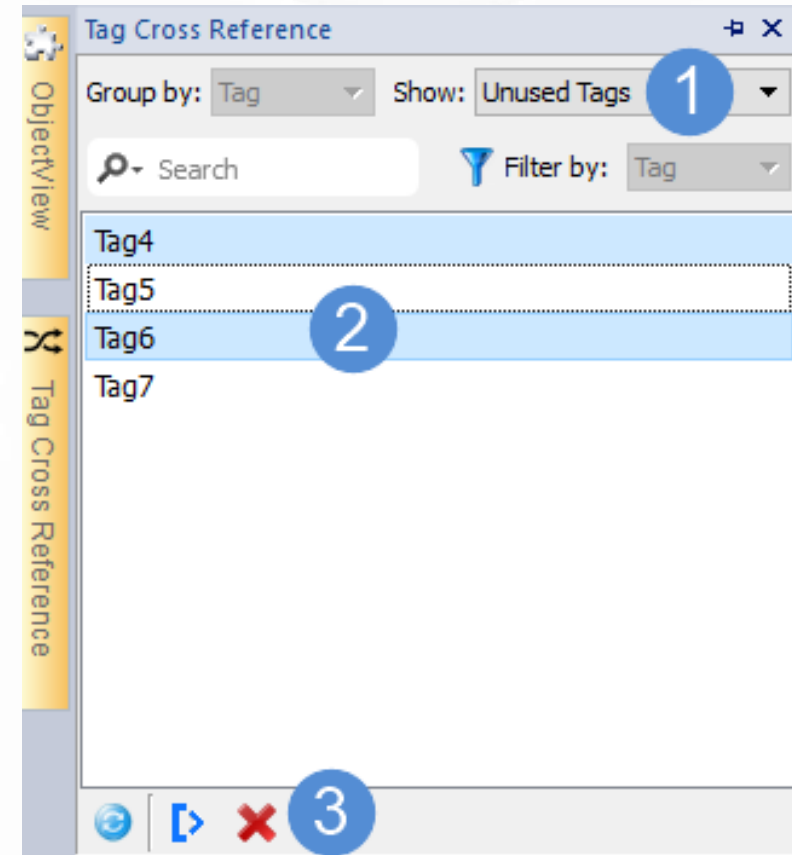


Tag Cross Reference pane

Unused Tags can be deleted to cleanup project in final step

1. Filter by “Unused Tags”
2. Select Tags to be deleted
3. Press red “X” icon

Note: Tags can be used in JavaScript by composing Tag name dynamically, within code cycles. Such Tags are listed as “Unused Tags”. Pay attention on deleting these Tags.



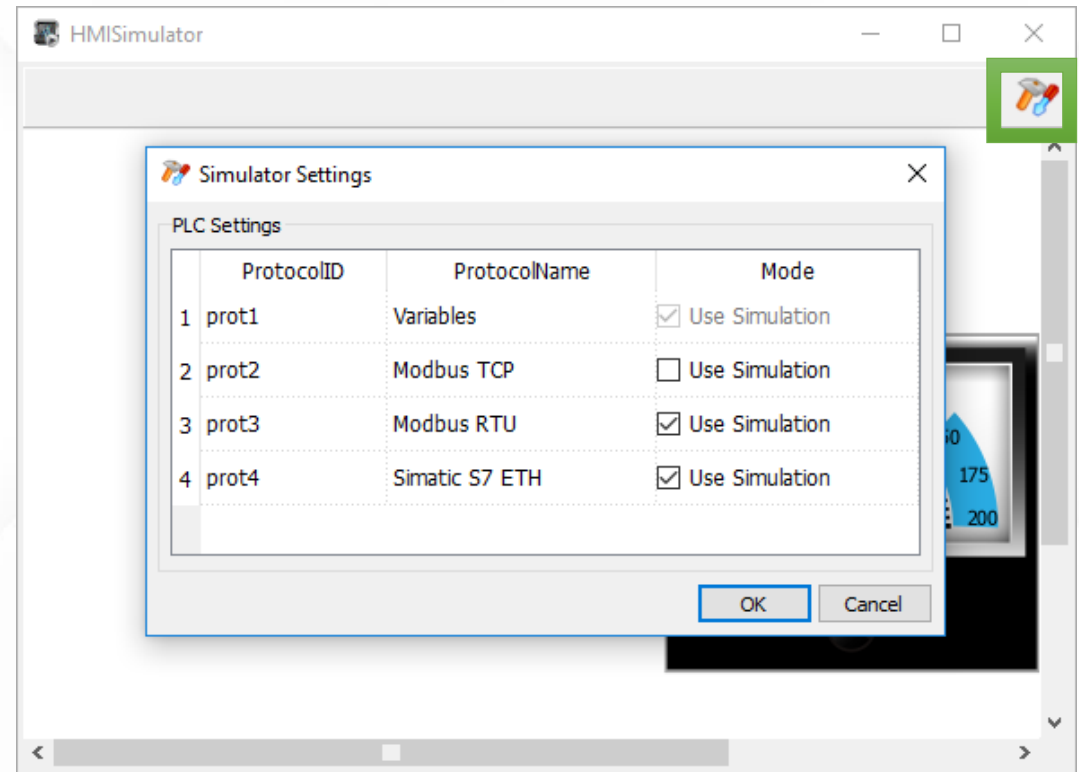
Simulator

Allows testing projects

Speeds up development

Supports

Simulation mode: default
Online mode: unchecking
"use simulation" for Ethernet
or serial RS-232 protocols



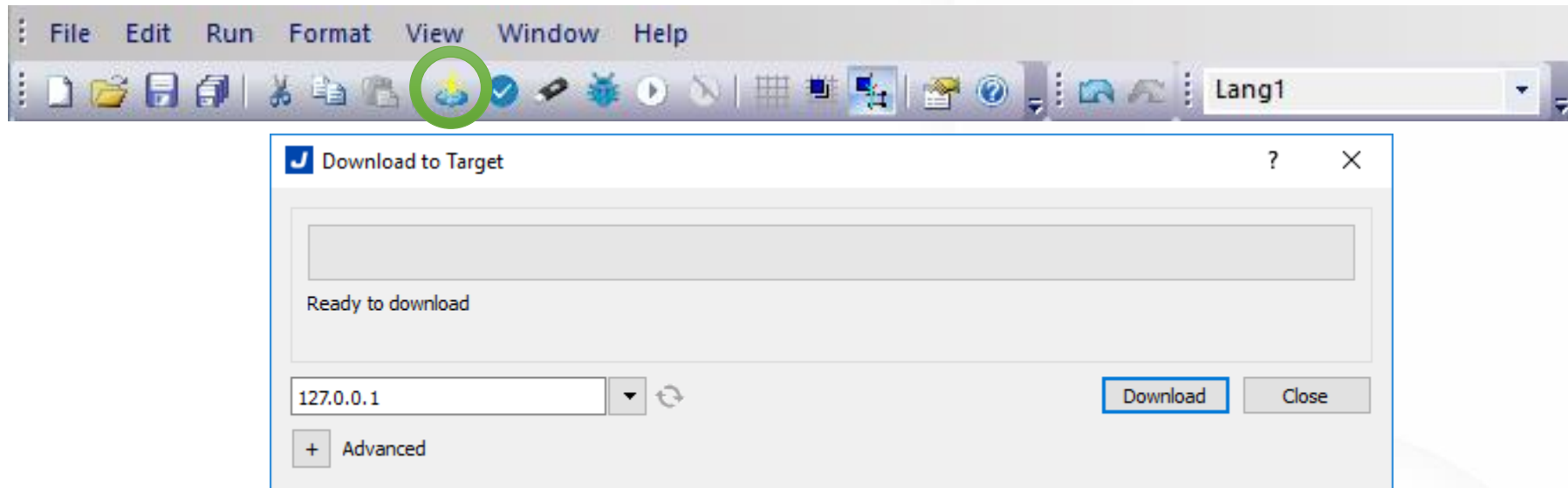
Ethernet Project Download

To download via network click on "Download to Target" icon



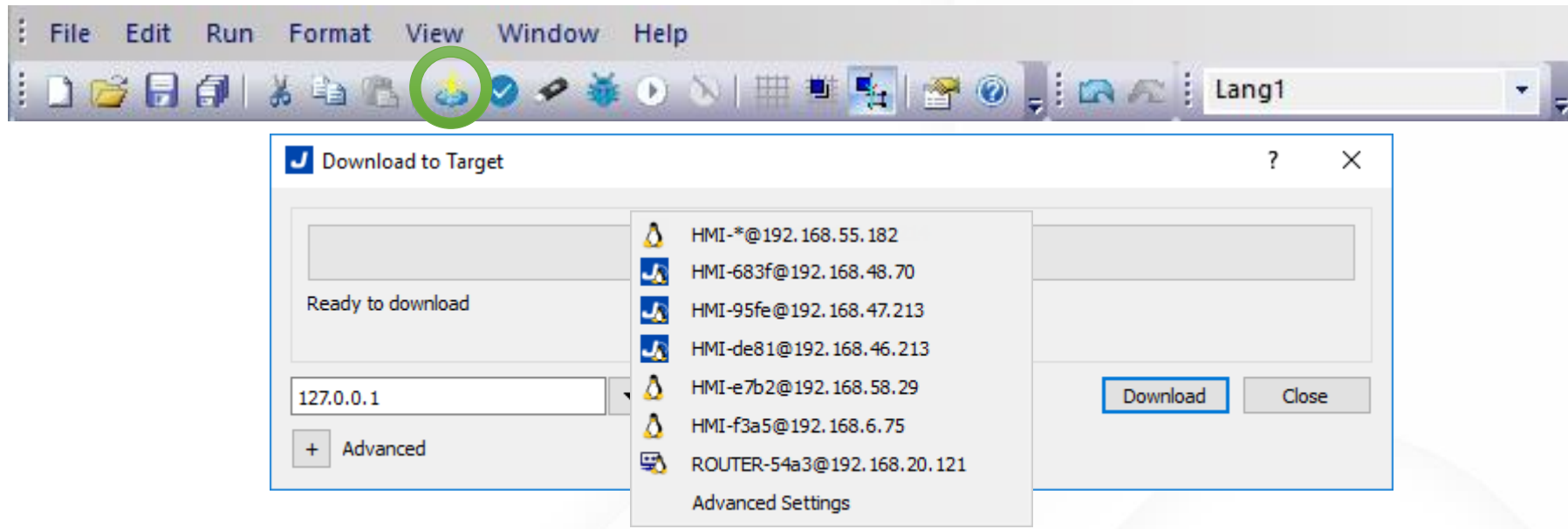
Ethernet Project Download

To download via network click on "Download to Target" icon



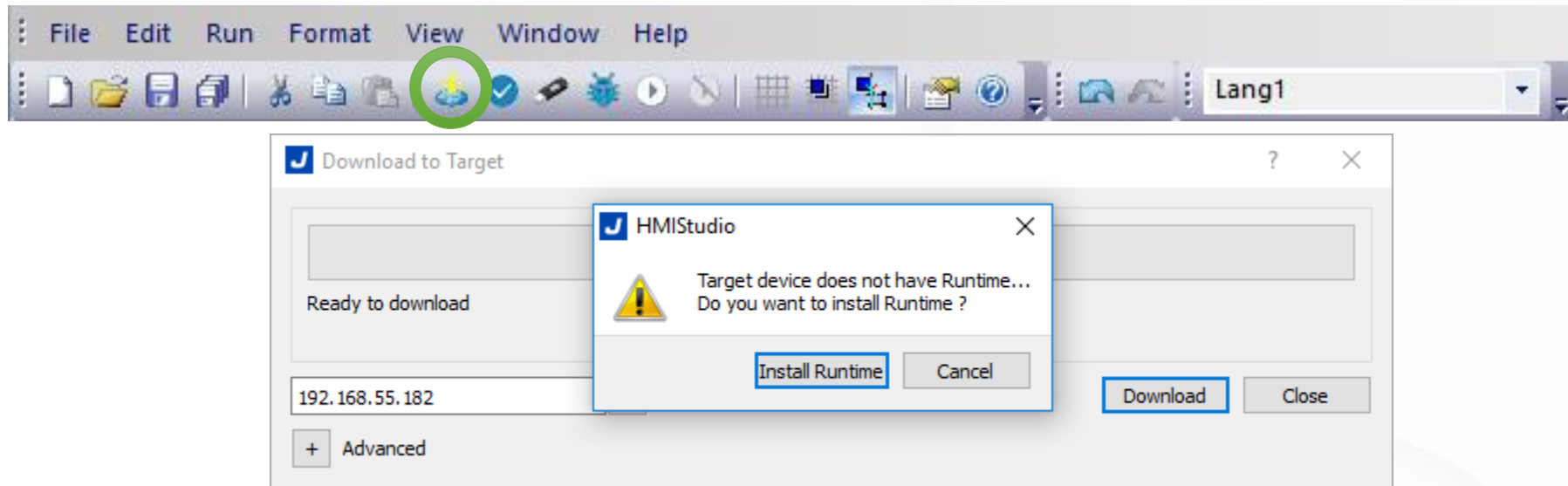
Ethernet Project Download

To download via network click on "Download to Target" icon



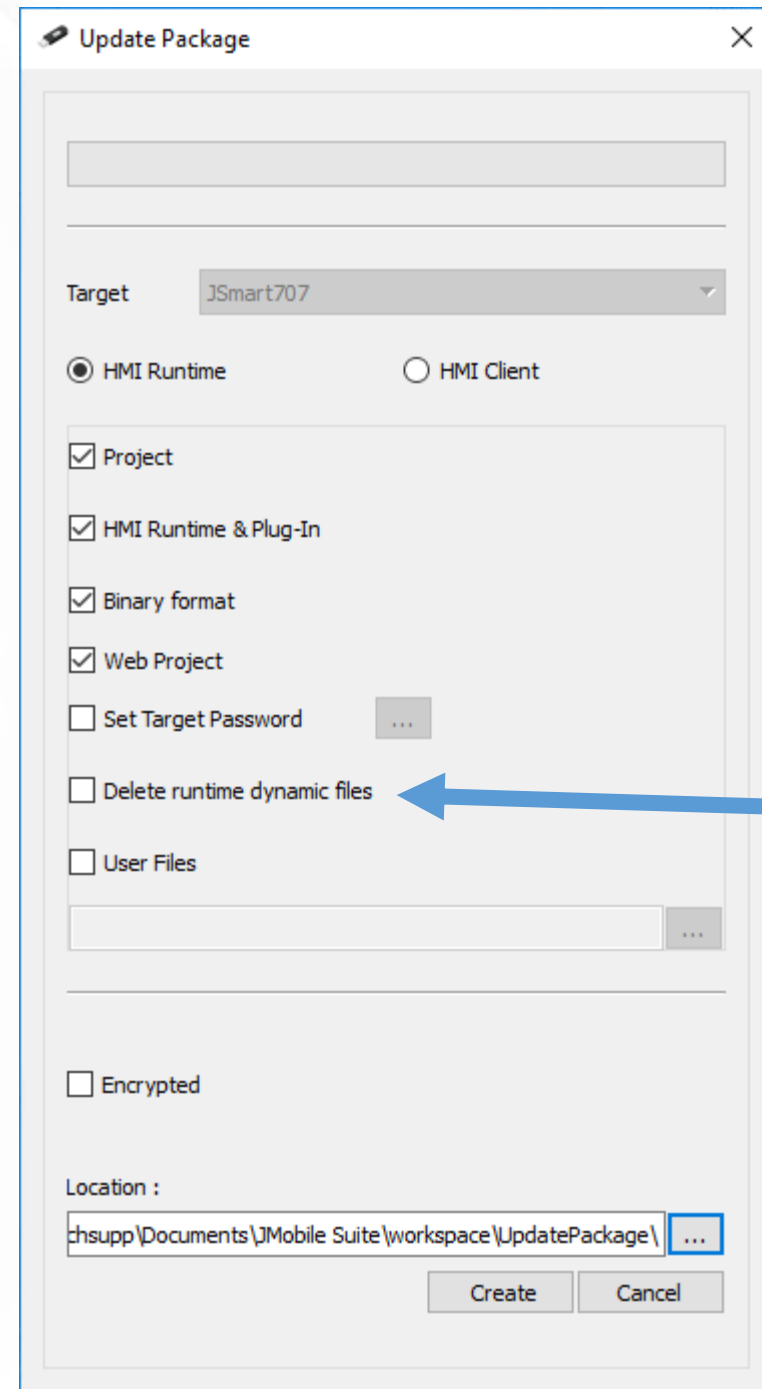
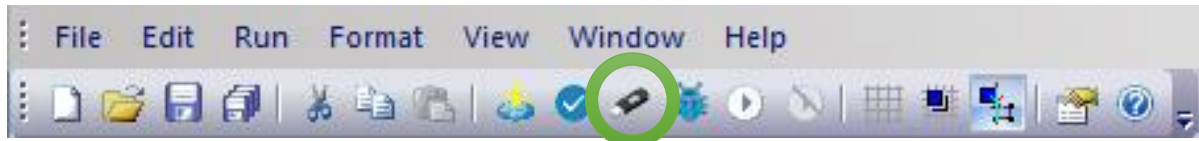
Ethernet Project Download

To download via network click on "Download to Target" icon



USB Project Download

To download via USB Memory:
click on "Update Package" icon

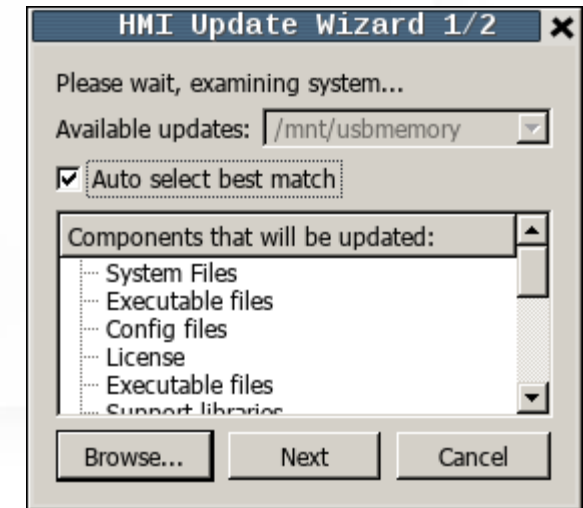
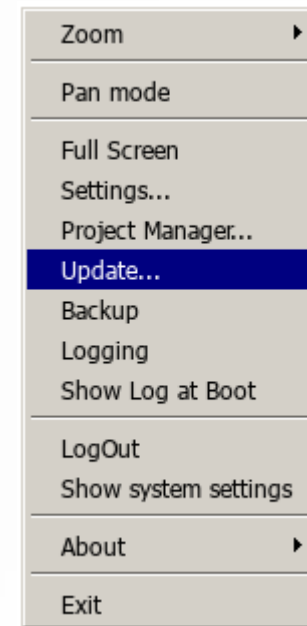
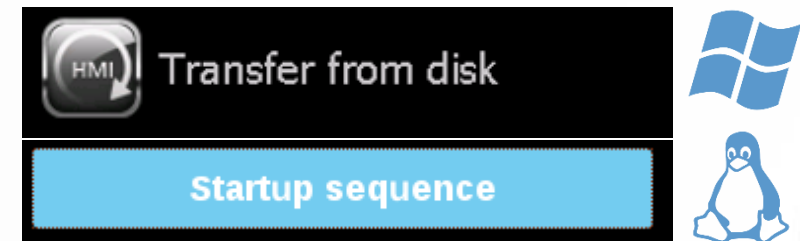


NEW
in 4.0

USB Project Download

How to deploy "Update Package" into HMI, via USB:

1. When Runtime is NOT present:
 - Click on "Transfer from disk" (WCE)
 - Click on "Startup sequence > Install" (Linux)
2. When Runtime is installed on HMI:
 - Touch and hold-on the finger on screen to get the context menu
 - Select "Update..."
 - Click on "Next"



HMI Client

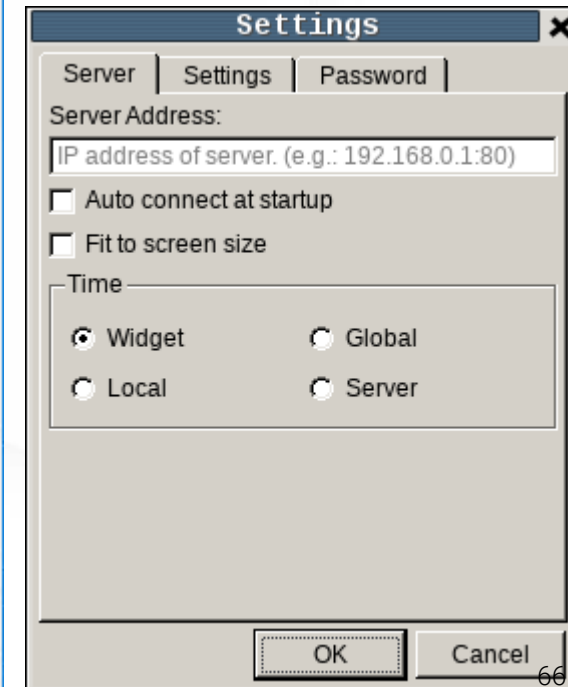
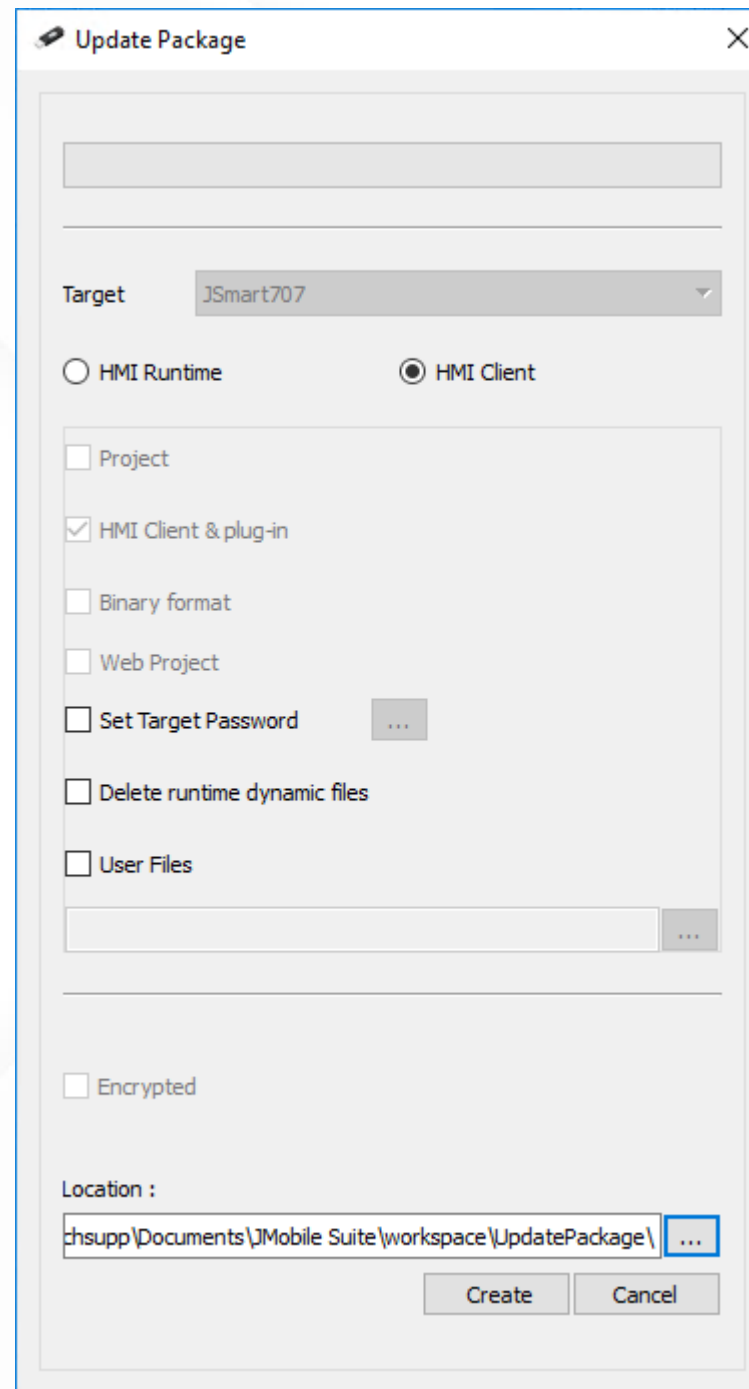
Available only on Linux HMIs 

Steps:

Create Update Package > HMI Client

Install it via USB

Connect to existing HMI



HMI Client

Example schema

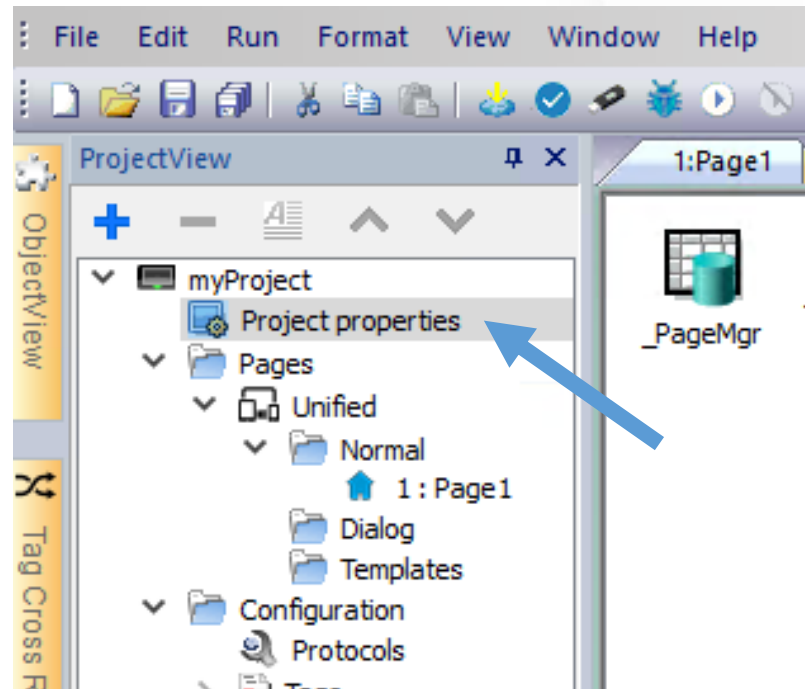


Project Properties

Allows access to project settings using by the Properties Pane

Common settings:

- Context menu protection
- Buzzer on touch
- Home page
- Page change from Tags



Properties

Project Widget : myProject	
Id	myProject
Full Path	C:\Users\techsupp\Documents\jmobile suite\w
Version	
Project GUID	3B6E0F17-9996-4408-8DDB-531F9BC7FC82
Runtime	
Context Menu	on delay
Developer Tools	false
Buzzer on touch	false a +
Buzzer duration (ms)	200
Keyboard	true
JavaScript Debug	false
Allow JavaScript Remote Debugger	false
Image DB Enable	true
FreeType Font Rendering	true
Communication icon delay (ms)	0
Fastboot mode	Default
Plug-in	
WebKit	false
Text Editor	false
Project	
Project Widget : myProject	
JS properties:	
<i>objectName</i>	
<i>id</i>	
<i>x</i>	
<i>y</i>	
<i>width</i>	
<i>height</i>	
<i>visible</i>	
<i>opacity</i>	
<i>rotation</i>	
<i>value</i>	
<i>userValue</i>	
<i>disable</i>	

Create Pages

Right click on Pages folder and
Insert new page

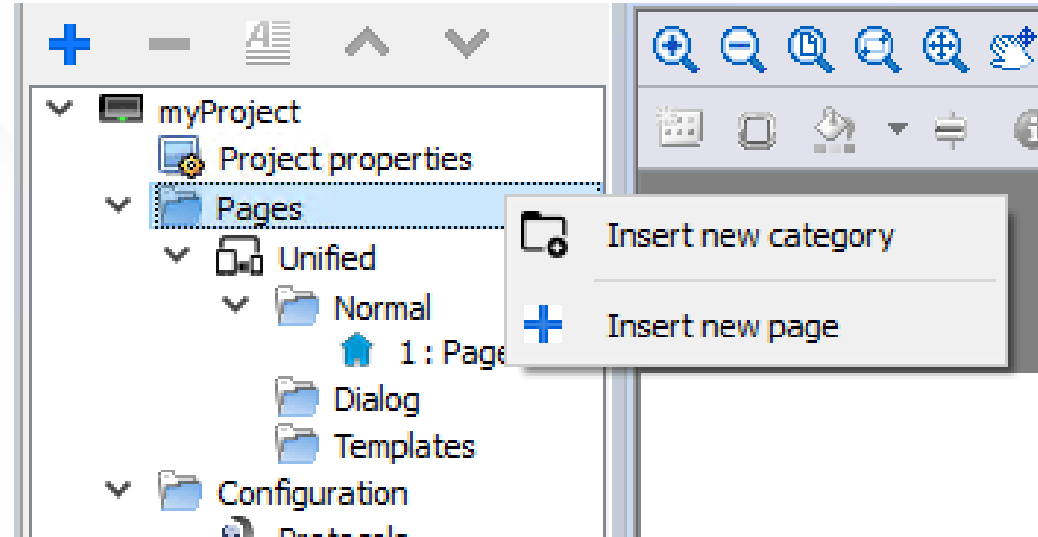
Possibility to organize categories with
specific access technology

NEW
in 4.0

Multi-selection treeview

Any category can be sub-organized in groups

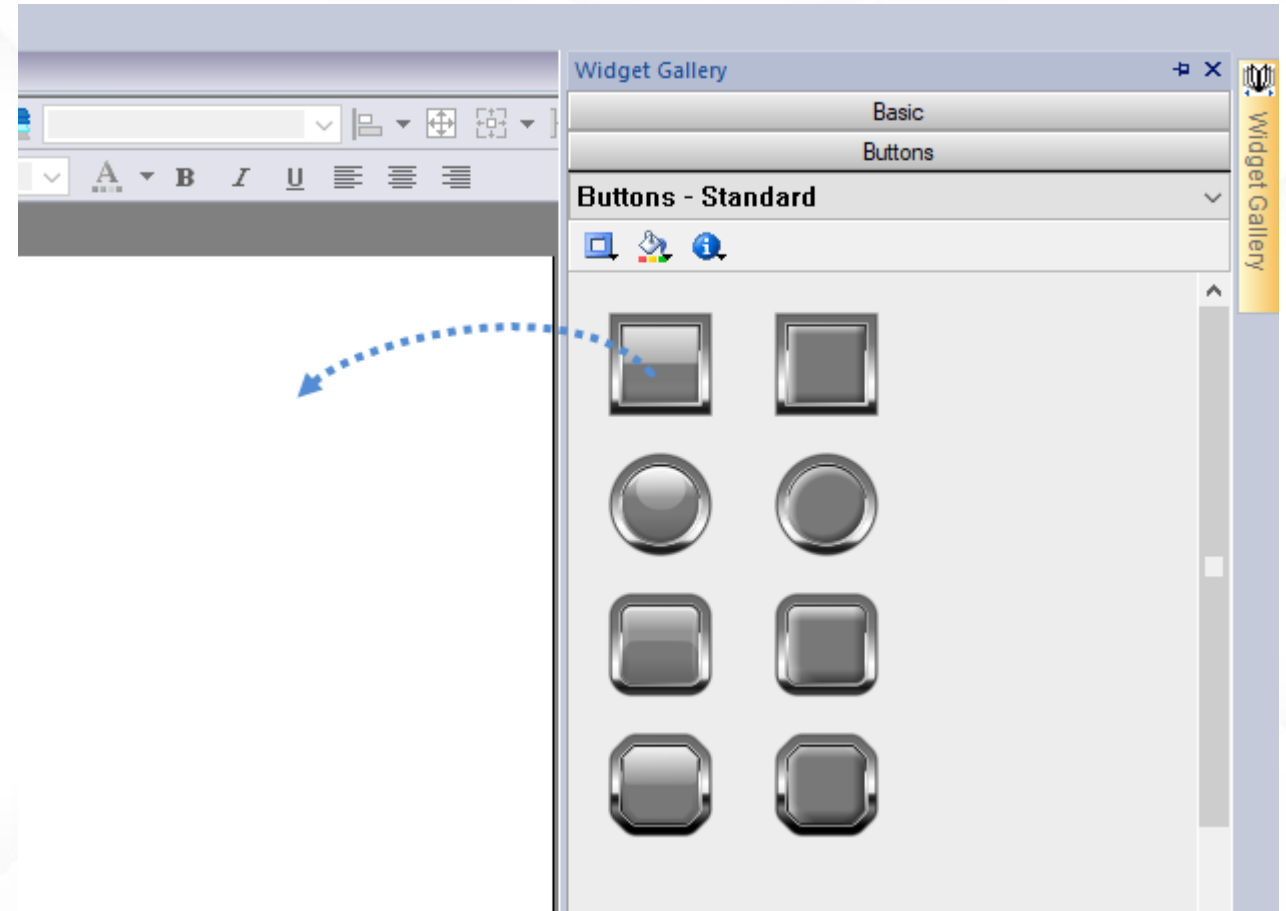
Possibility to import pages from other project



Widget Gallery

Drag and drop widgets in page

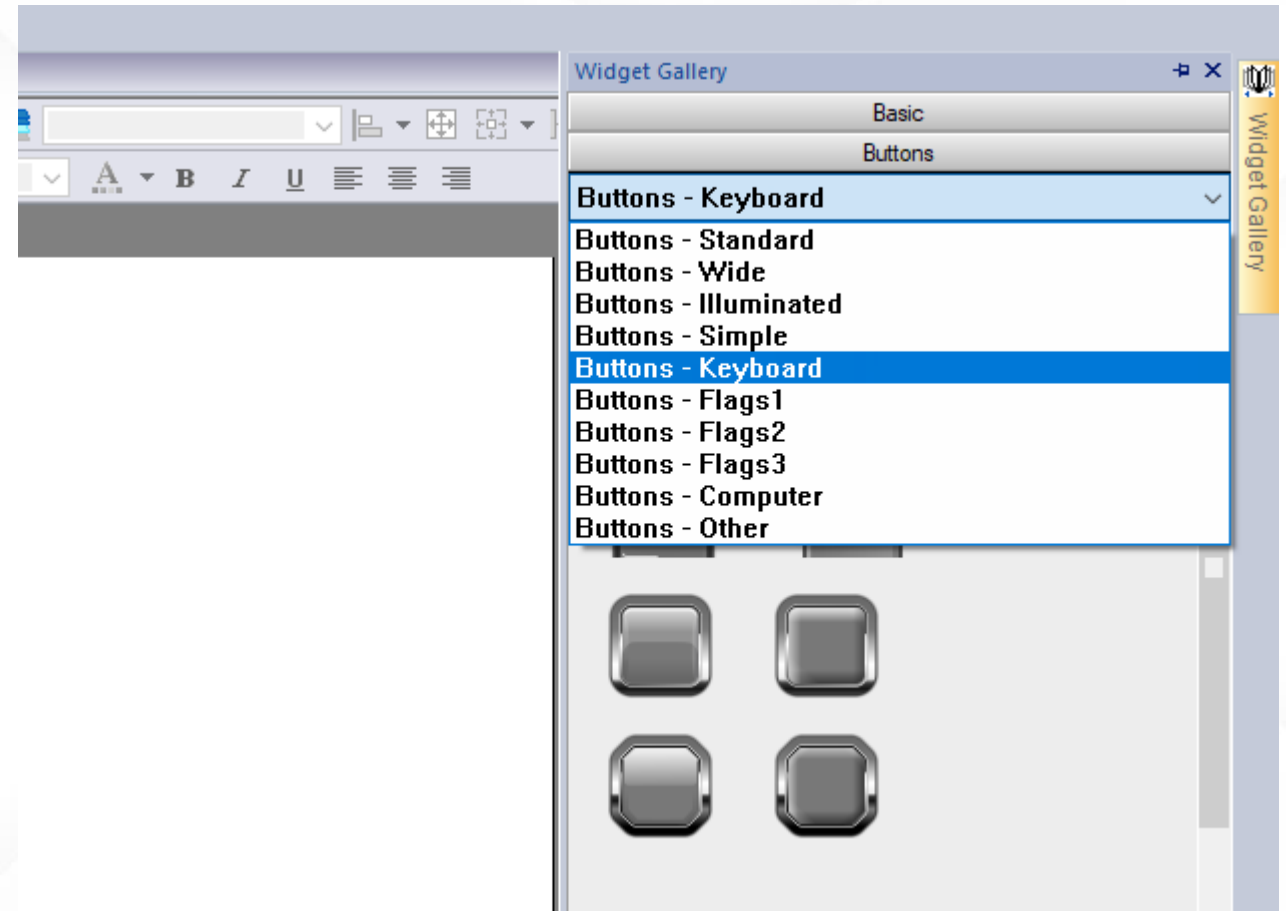
Divided by category and sub-categories



Widget Gallery

Drag and drop widgets in page

Divided by category and sub-categories



Using Dialog pages

Dialog pages: pop-up windows opened at runtime on top of current page

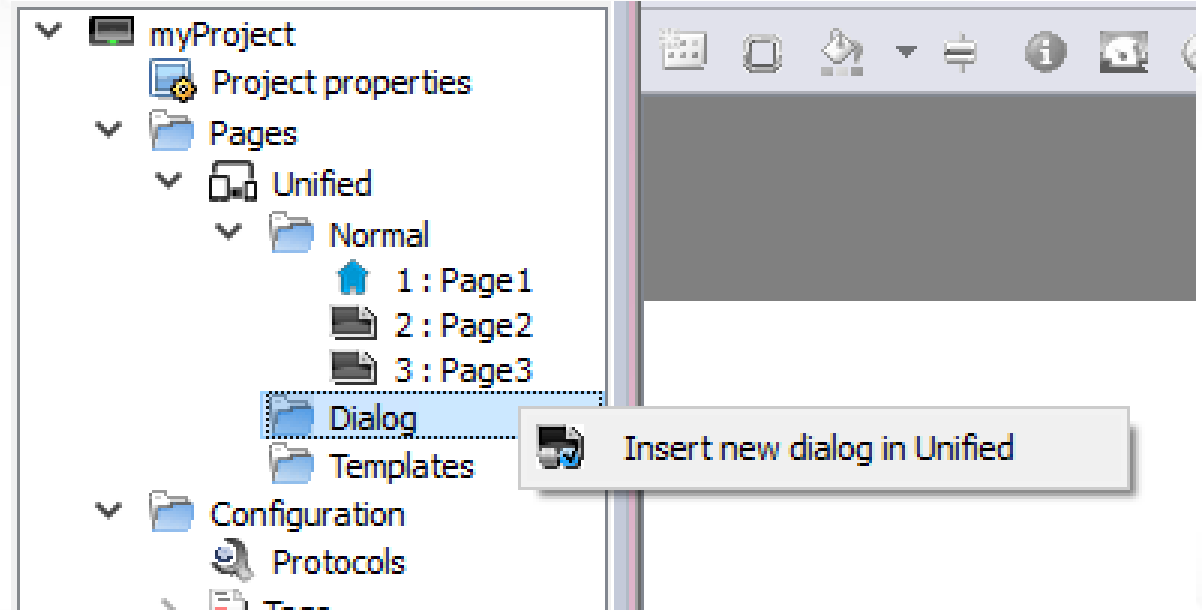
Dialog type can be:

Modal: the focus is only on dialog,

Non-Modal: user can continue to use main project window or other non-modal dialogs.

Runtime position can be customized

Up to 5 dialog pages can be opened at the same time

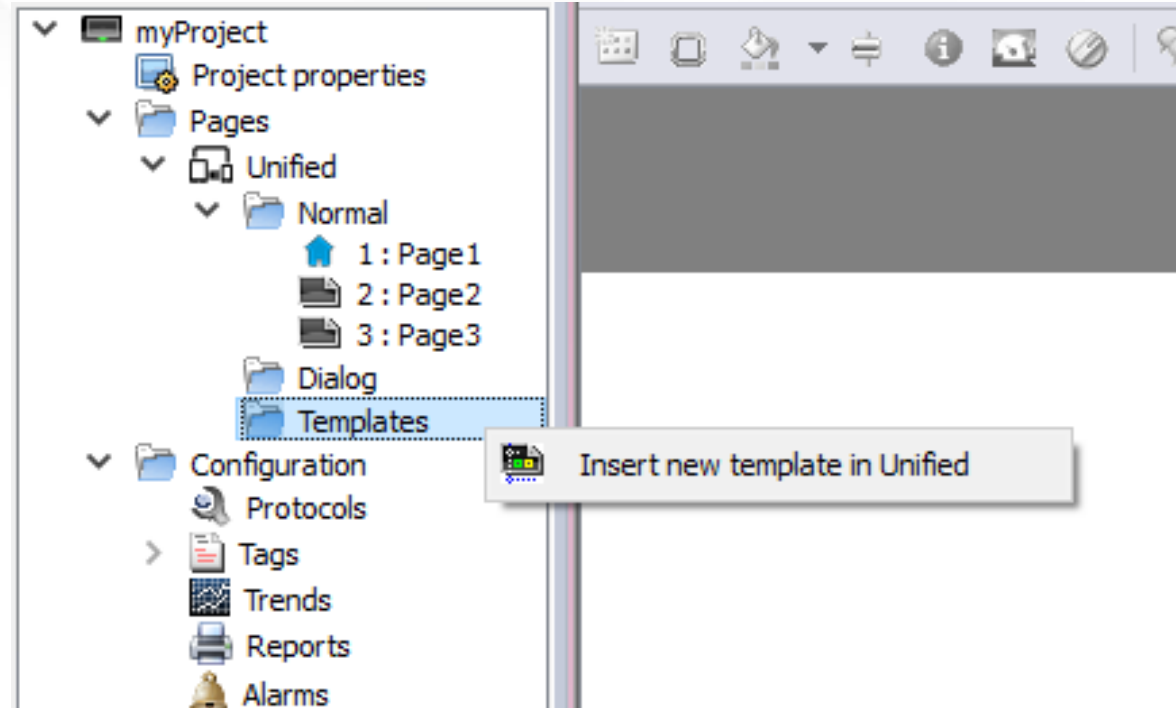


Using Template pages

Constant elements for pages like Header and Footer

Several templates can be defined

Different pages can have different templates



Using Template pages

Constant elements for pages like Header and Footer

Several templates can be defined

Different pages can have different templates

Page : Page1	
Id	Page1
Width	800
Height	480
Background	<input type="checkbox"/> [255, 255, 255] a +
Template	none <input type="button" value="v"/>
Static Optimization	none
Static File Type	TemplatePage1 png
JavaScript Debug	false
Keyboard	true
Precache	false
Events	
OnActivate Action	<input type="button" value="+"/>
OnDeactivate Action	<input type="button" value="+"/>
OnWheel Action	<input type="button" value="+"/>

Protocols

Configuring protocol

Open Protocol editor from ProjectView

Add a communication driver:

- Click [+]

- Select from the list

- Configure the driver

Supports up to *

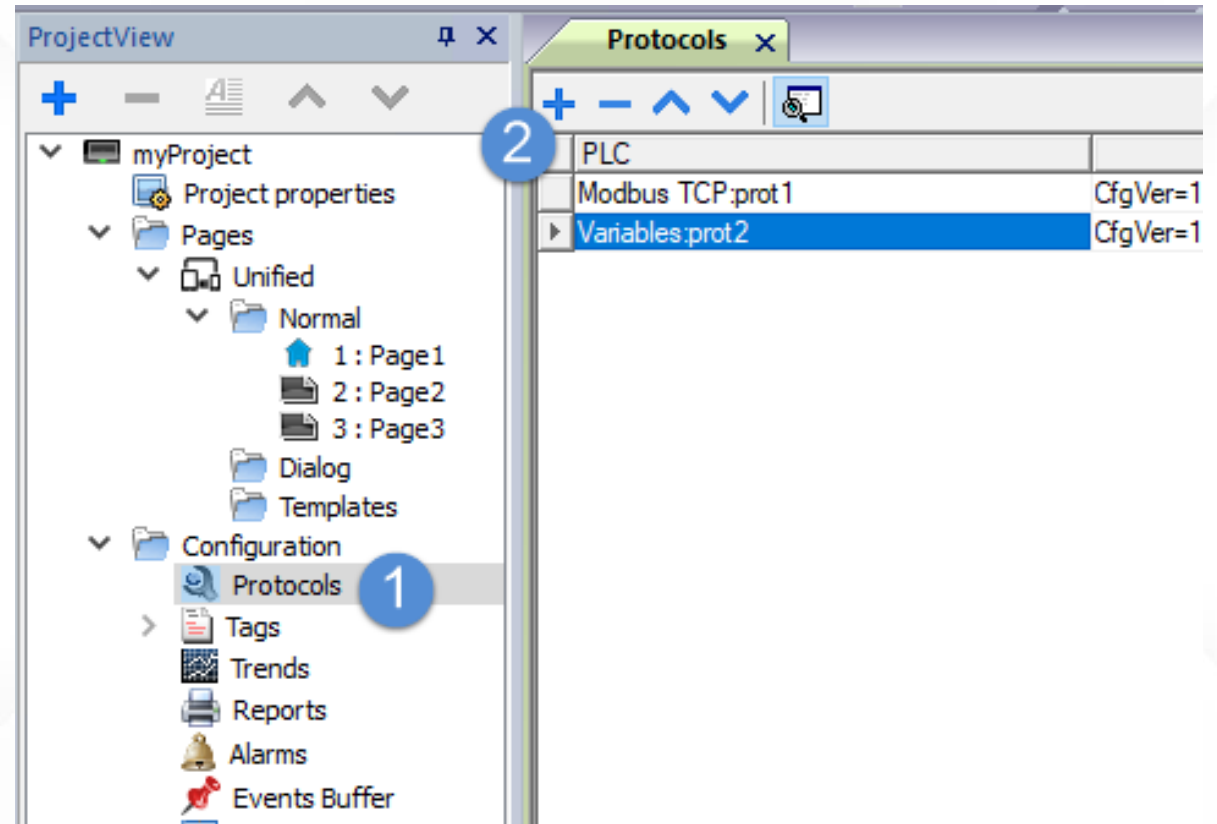
- 4 protocols for single core CPU HMIs

- 8 protocols for multicore CPU HMIs

Virtual protocols

- Variables

- System Variables

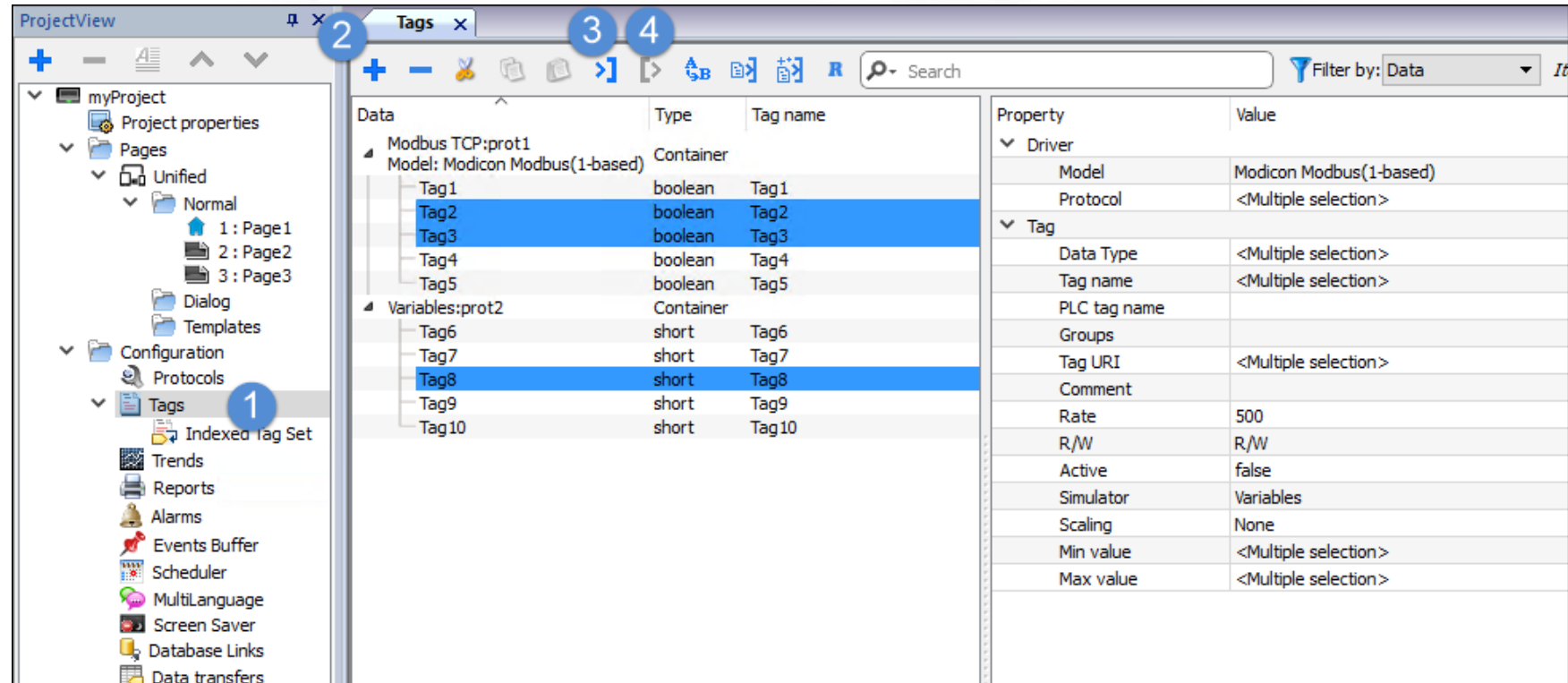


* Except for Virtual protocols

Tags and Indexed Tag Sets

Configuring Tags

1. Open the Tag editor from ProjectView
2. Manually add Tags...
3. ...or Import Tags from PLC symbol file *
4. Export Tags export in xml format



NEW
in 4.0

Organized in treeview with multiple selection and editing
 Quick toolbar for search and check number of defined Tags
 * Unified view between dictionary Tags and imported Tags

Tag Find and Rename

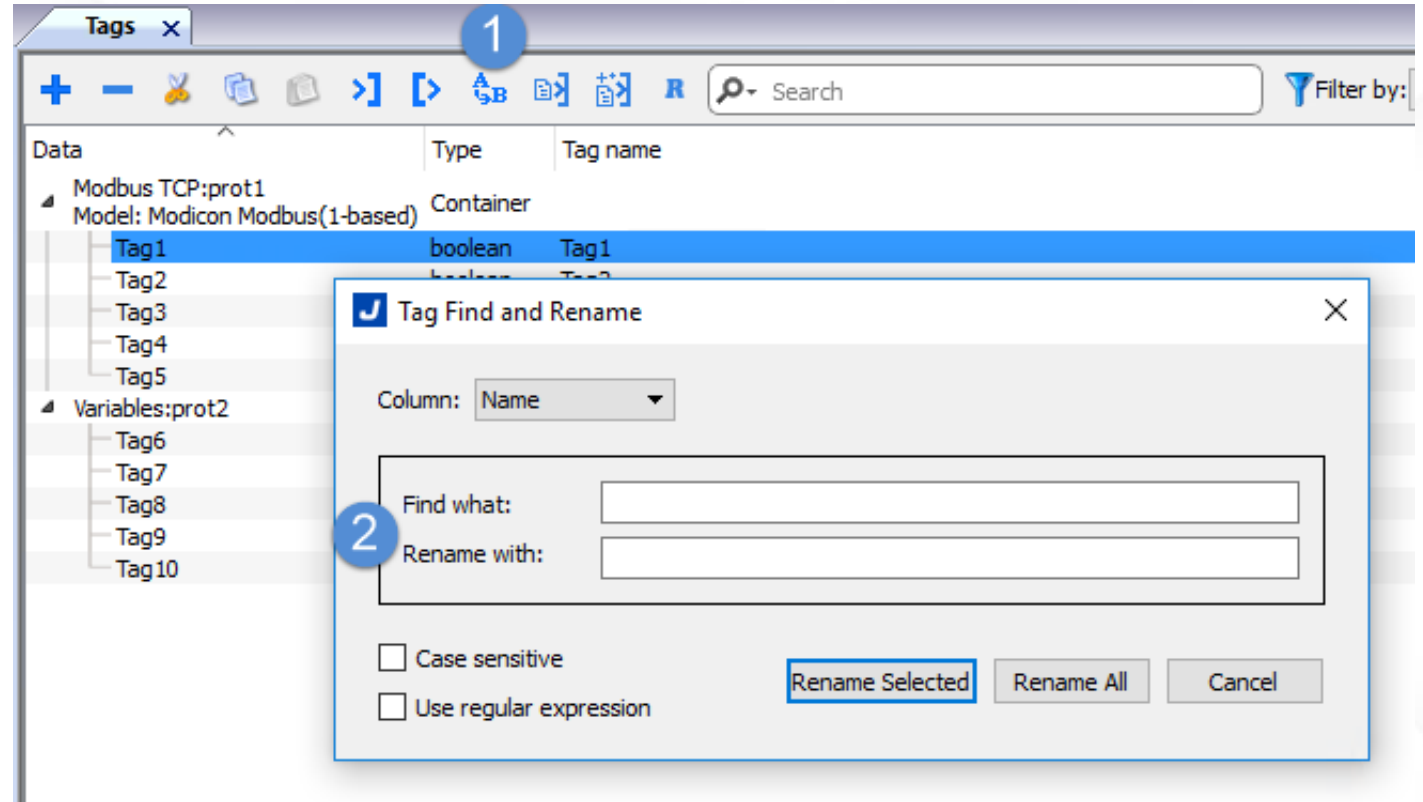
Rename many Tags at same time

Click on Find and Rename icon

Choose what to find and rename


Change common properties with one click

Using Column filter



Tag data types in JMobile

“Short” is 16-bit data
corresponds to
“INT” in IEC 61131



JMobile Data Type	Limits			Memory Space
Boolean	0	...	1	1 bit data
Byte	-128	...	127	8-bit data
UnsignedByte	0	...	255	8-bit data
Short	-32768	...	32767	16-bit data
UnsignedShort	0	...	65535	16-bit data
Int	-2.1e9	...	2.1e9	32-bit data
UnsignedInt	0	...	4.2e9	32-bit data
Int64 *	-9.2e18	...	9.2e18	64-bit data
UnsignedInt64 *	0	...	1.8e19	64-bit data
Float	1.17e-38	...	3.40e38	corresponds to IEEE 754 single-precision 32-bit floating point type
Double	2.2e-308	...	1.79e308	corresponds to IEEE 754 double-precision 64-bit floating point type

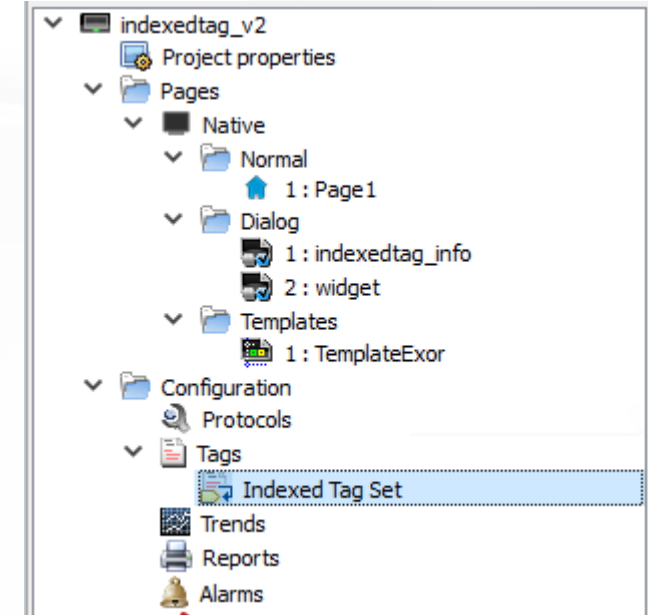
* Availability depends on specific protocol

Indexed Tag Sets

Allow to dynamically attach a tag using an alias

Useful to duplicate same object but with different tags attached

Example: we have three engines, each one have **rpm** and **oil**



Data	Type	Tag name
Variables:prot1	Container	
dialog_opened	boolean	dialog_opened
engine_index	unsignedByte	engine_index
engine1_oil	unsignedShort	engine1_oil
engine1_rpm	unsignedShort	engine1_rpm
engine2_oil	unsignedShort	engine2_oil
engine2_rpm	unsignedShort	engine2_rpm
engine3_oil	unsignedShort	engine3_oil
engine3_rpm	unsignedShort	engine3_rpm



rpm	oil
999	999

Indexed Tag Sets

Defining an Indexed Tag Set called "engine", driven by tag "engine_index", allow me to "attach" the Alias "oil" or "rpm" instead of real tag

Use one "engine widget" instead of three

The screenshot shows the configuration for an indexed tag set named "engine". At the top, there are "Add" and "Delete" buttons. Below that, the tag name "engine" is entered in a field, and "engine_index" is selected as the "Index Tag". A search bar and a "Filter by: Index" dropdown are also visible. Below the search bar, there are controls for "Index Instance" and "Alias". The main part of the interface is a table with three columns: "Index", "rpm", and "oil". The "Index" column has values 1, 2, and 3. The "rpm" column has values engine1_rpm, engine2_rpm, and engine3_rpm. The "oil" column has values engine1_oil, engine2_oil, and engine3_oil.

	Index	rpm	oil
1	1	engine1_rpm	engine1_oil
2	2	engine2_rpm	engine2_oil
3	3	engine3_rpm	engine3_oil

The screenshot shows the configuration for a field named "field7.value". At the top, there are radio buttons for "Source": Tag, Alias, System, Widget, and Recipe. The "Alias" radio button is selected. Below that, there is a search bar. The main part of the interface is a tree view showing the field "engine" with two sub-items: "oil" and "rpm". The "rpm" item is highlighted in blue.

Trends and Data logging

Trends

Configuring "Trend buffers"

- Tags selection

- Sampling conditions

 - Time or Trigger

- Sampling filter

 - Mechanism to save space in memory

Configuring "Trend widgets"

- History trends

 - Connected to trend buffer

- Real time trends

 - No data storing, connected to Tag

Trend Buffers

Open the Trend editor and add a trend

Configure buffer options

Tags to be sampled and format *

Sampling Time *

Number of Samples

Storage Device

Trigger

Blank: disabled

Tag: allows to sample "on event"

Sampling Filter

* = new options



The screenshot shows the 'Trends' editor window. At the top, there are buttons for 'Add trend', 'Add PLC trend', 'Delete trend', and 'Settings'. A 'Total memory Space' indicator shows 3.3% usage. The main configuration area is for 'Trend1', which is active and has the source 'engine_1_oil'. The configuration options include:

- Number of Samples:** 43000
- Timestamp:** Use source timestamp
- Sampling time:** 60
- Time:** Sec, 1/10 Sec
- Trigger:** None
- Sampling Filter:**
 - Current Sample value - Previous Sample value <: 0.00
 - Current Sample value - Previous Sample value >: 0.00
- Storage Device:** Local, USB, SD, Preferred
- Path:** Data/
- Backup Archive:** Save a copy when full, .csv
- Storage Options:** Local, USB, SD, Preferred
- Path:** Data/
- Select Fields:** 0;5;4
- Select Curves:** All curves
- Time Spec:** Local
- Date Format:** [Dropdown]
- Language:** [Dropdown]

At the bottom, there is a table with the following columns: Name, Title, Tag, Format, Comment.

	Name	Title	Tag	Format	Comment
1	Name1	myTitle	engine1_oil	Numeric	

Trend Buffers

Automatically save a CSV file containing trend data, when trend buffer gets full

Save in external memory (USB or SD)

Specify time settings

Specify destination path

	Name	Title	Tag	Format	Comment
1	Name1	myTitle	engine1_oil	Numeric	

Trend Widgets

Basic category > Trends/Graphs

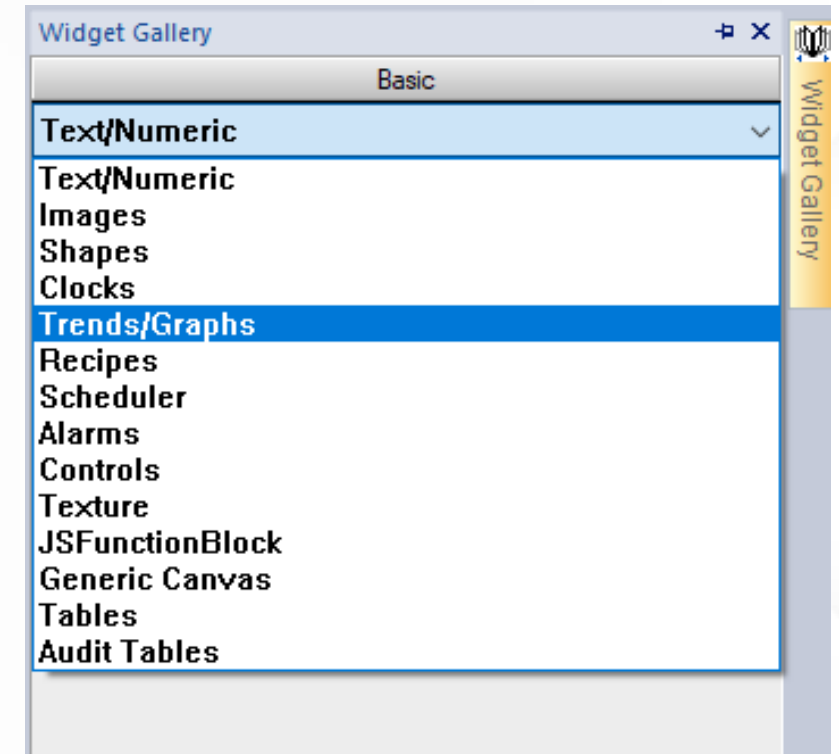
History Trend > attach curve to Trend buffer

RealTime Trend > attach curve to a Tag

Trend widgets support gestures

pan to move (for all series)

pinch to zoom (only for multitouch series)



Trend Widgets

Basic category > Trends/Graphs

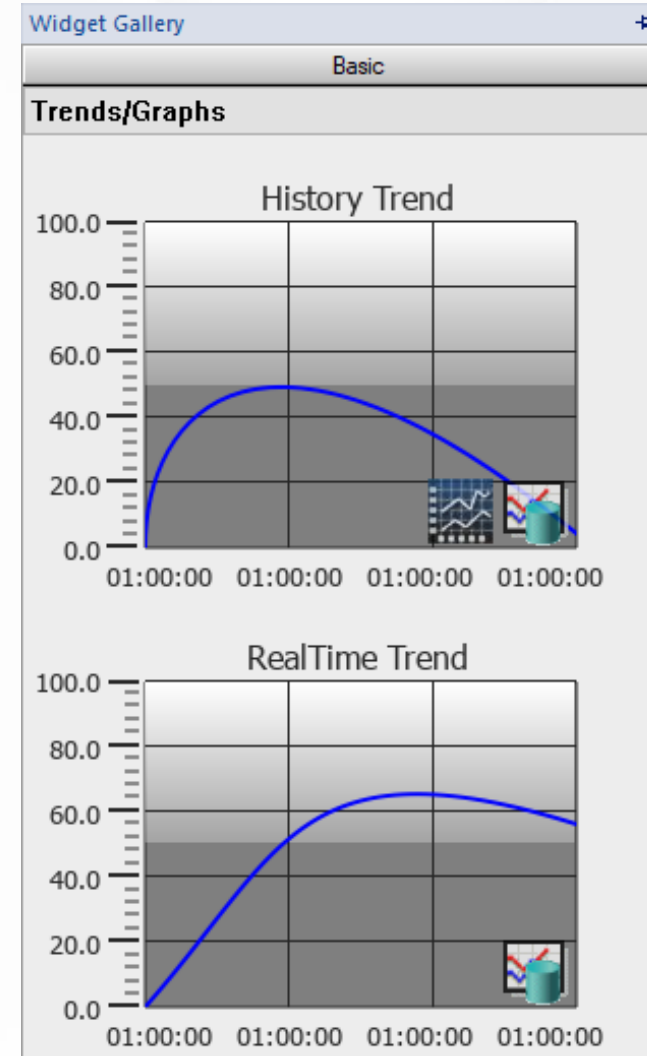
History Trend > attach curve to Trend buffer

RealTime Trend > attach curve to a Tag

Trend widgets support gestures

pan to move (for all series)

pinch to zoom (only for multitouch series)



Trend Widgets

Basic category > Trends/Graphs

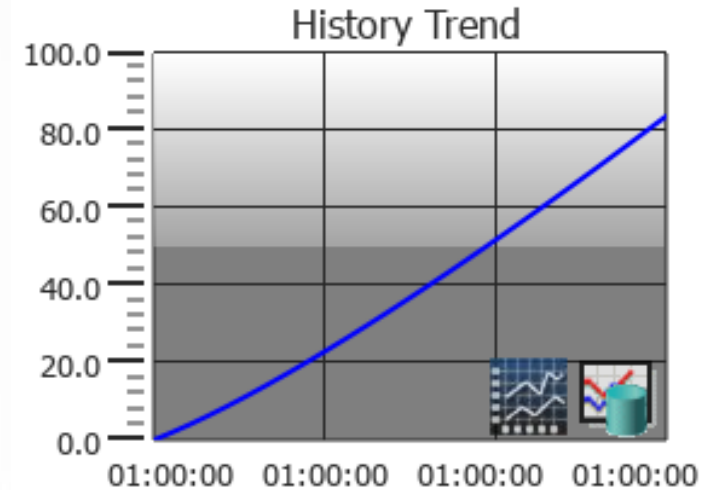
History Trend > attach curve to Trend buffer

RealTime Trend > attach curve to a Tag

Trend widgets support gestures

pan to move (for all series)

pinch to zoom (only for multitouch series)



Curve 1		
Curve 1 Trend		a +
Visible	tru	Select trend
Request Samples	10	Attach To...
MinY	n	

HistoricTrend.buffer1.Value

Source: Tag Alias System Widget Recipe

Search

Name

- HistoricTrend.trend
 - Trend1
 - Name1
 - Page Duration
 - Position

Trend Table Widgets

Basic category > Trends/Graphs

Trend Table > attach curve to
Trend buffer

Based on Table widget

View samples in numeric format

Trend Table

From: 08/02/18 - 10:06:48 Duration: 10 Mins Refresh

To: 08/02/18 - 10:06:48

Timestamp	Name1	Name2	Name3	Name4	Name5
Data	99999	99999	99999	99999	99999

Backward Forward

Exporting Trend Buffer

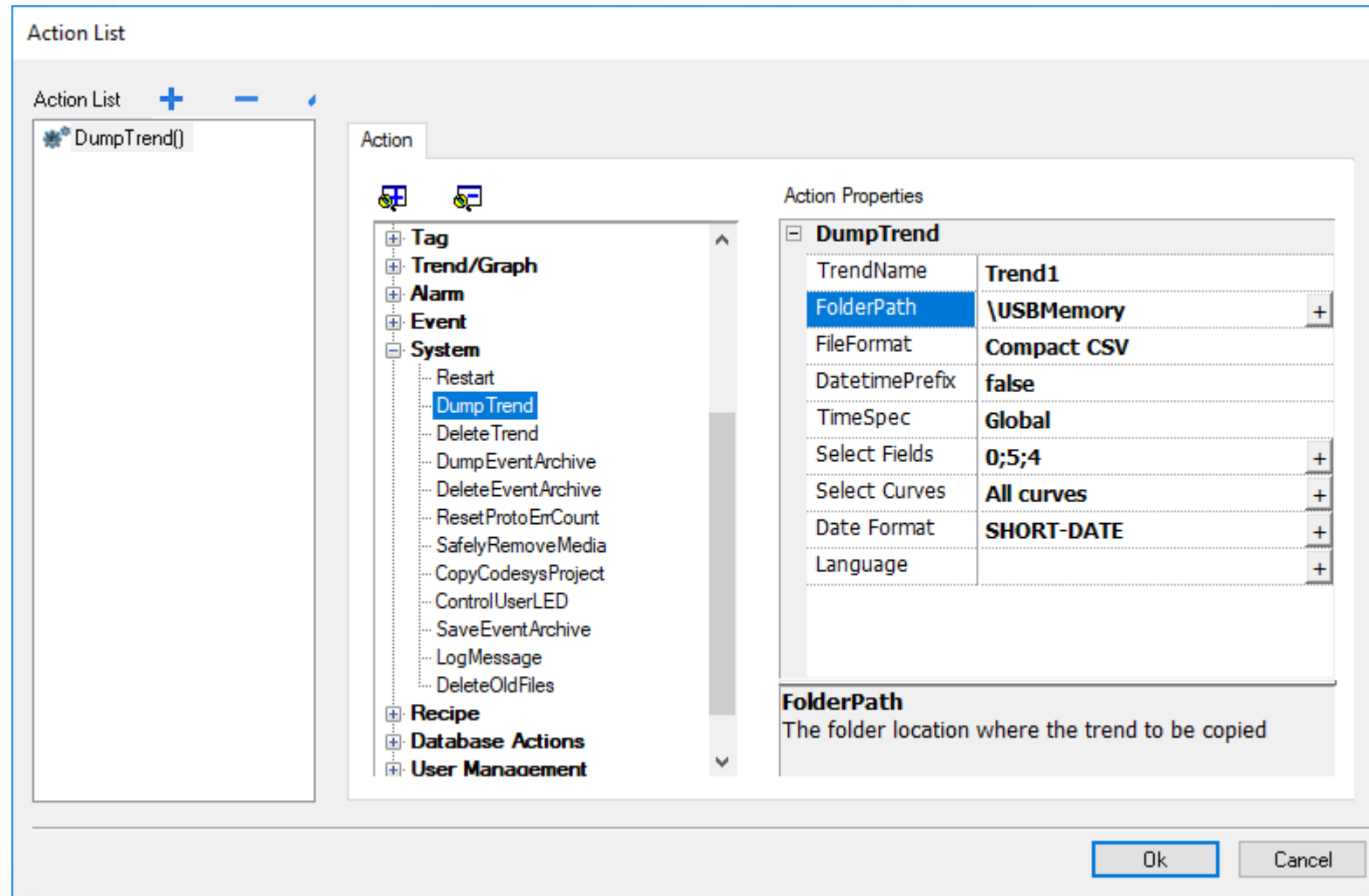
DumpTrend action

Save locally or on external memory

Save on Network folder

Select curves to export to CSV

Choose language



Reports

Reports

Allow printing Reports and Alarms

Supports USB printers *

Supports printing to PDF files

Report printing with report layout editor

Printing of Alarms on triggering
continuous printing *

Print screen image

* Available for WCE HMIs only

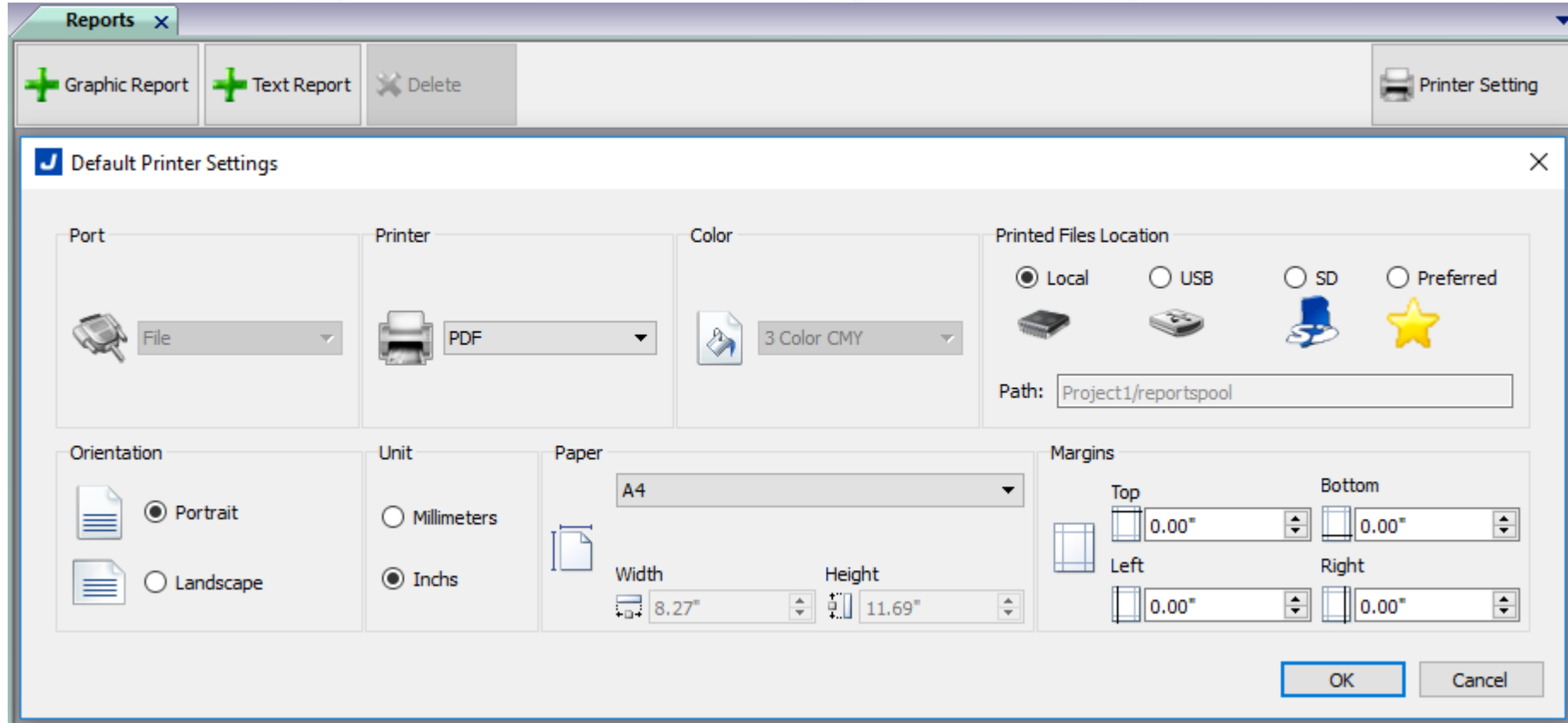


Reports

Reports can be defined as two types:

Graphic Report
creates layout with graphical objects

Text Report
real time printing of alarm events *



* Available for WCE HMIs only



Graphic Reports

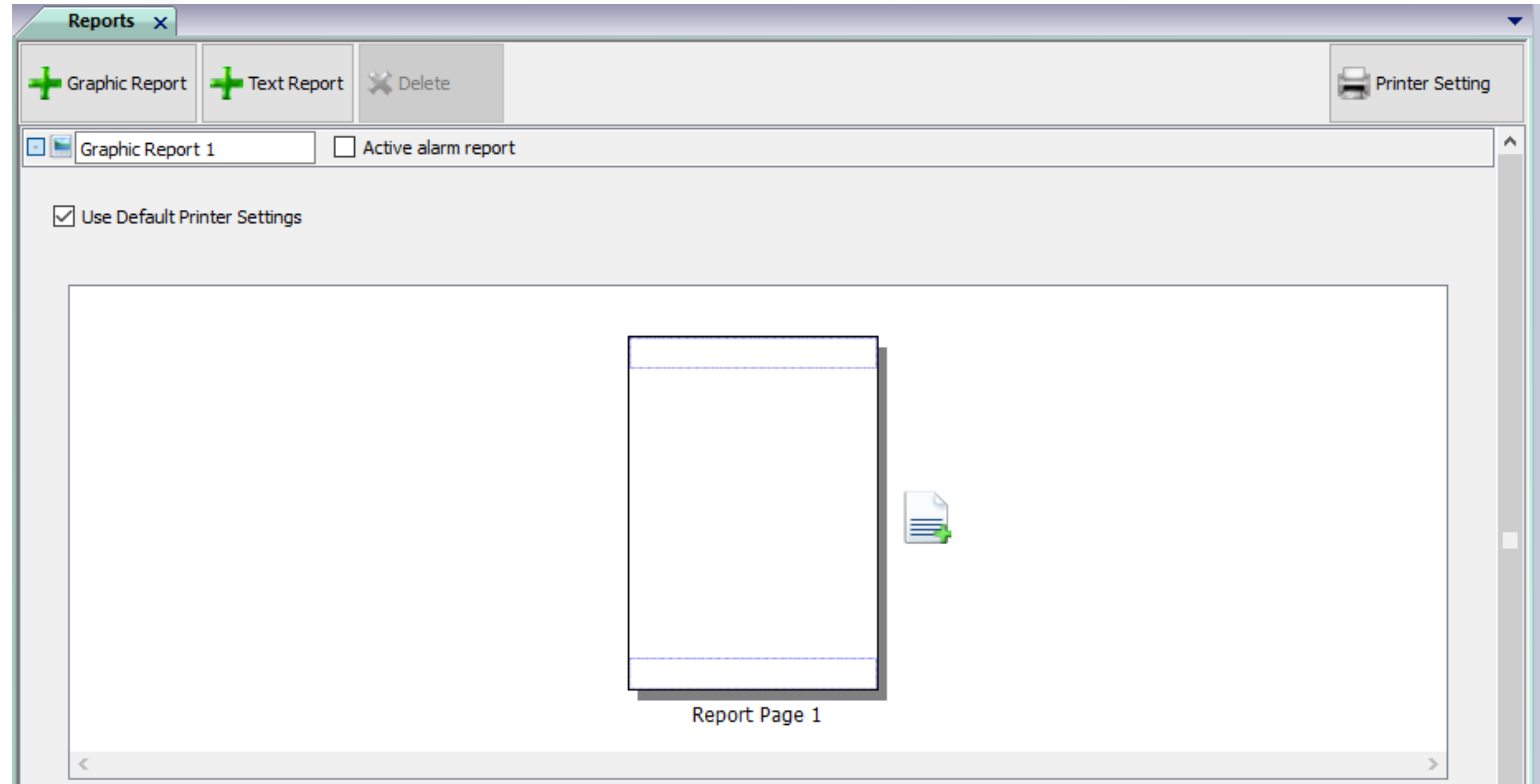
Graphic Reports can include:

Screenshot of current page shown on HMI

Images, labels and numeric fields

Alarm widget

Table widget

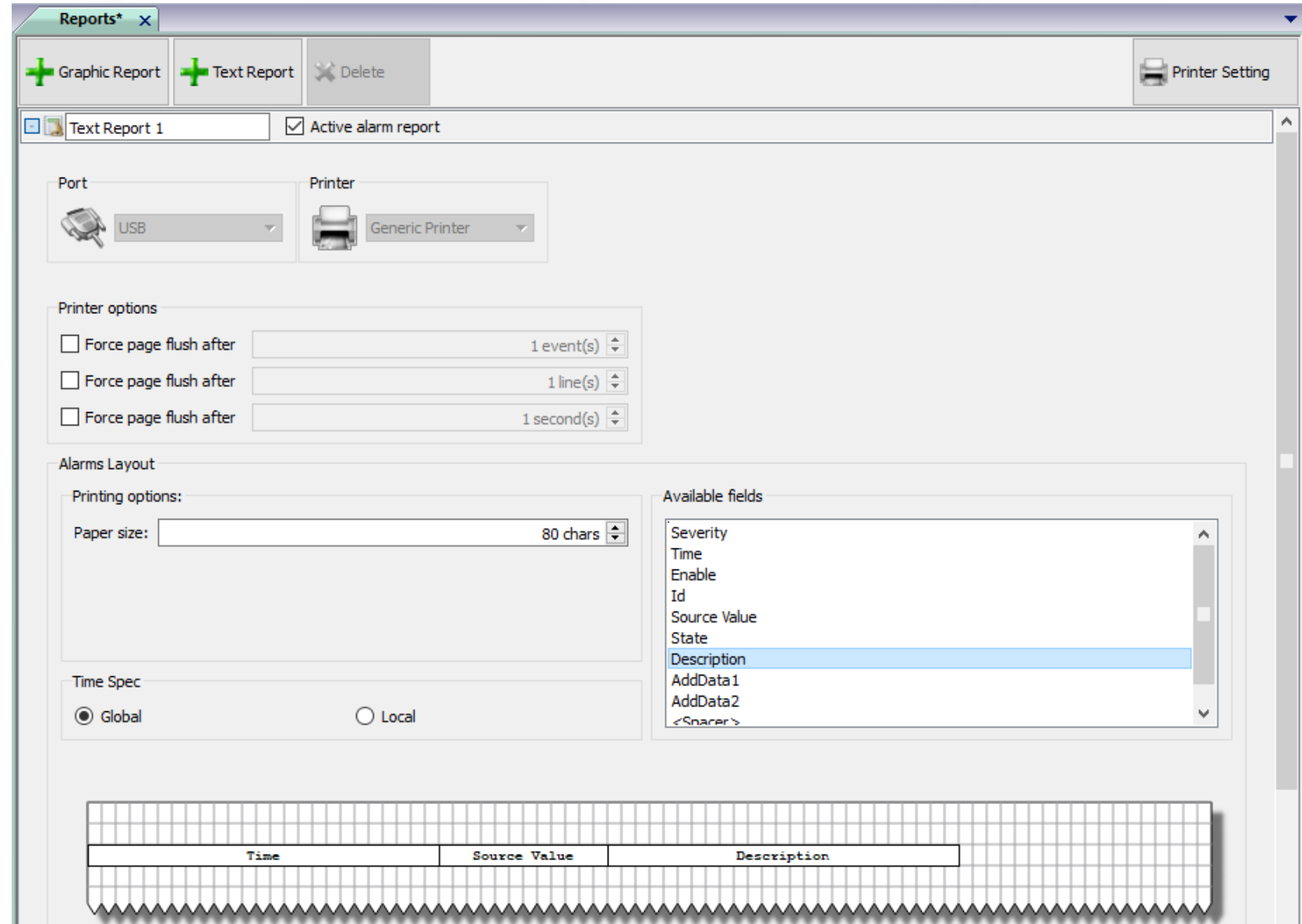


PDF files can be generated with electronic sign, in order to check file integrity

Text Reports

Is configured by default as an “Active alarm report”

Alarm Report cannot be printed in PDF format



Alarms

Alarms

Alarms concept

Configuring Alarms

- Triggers

- Actions

Alarm historical list

- Buffers

- Visualization of alarm history

Exporting Historical Event List

Live Tag values in alarm description

Alarms can be enabled/disable at runtime

Working with Alarms

Multi-selection treeview

Double click on any cell to edit alarms

NEW
in 4.0

The screenshot shows the EXOR software interface with the 'Alarms' configuration window open. The interface is divided into three main sections:

- Left Pane (Treeview):** Shows a hierarchical treeview of the project structure. The 'Alarms' folder is selected, indicated by a blue circle with the number '1'.
- Main Pane (Table):** Displays a table of configured alarms. A blue circle with the number '2' highlights the '+' icon in the toolbar above the table. The table has the following columns: Name, Groups, Enable, Ack, Trigger, Tag, Action, and Description. Eight alarms are listed, all with 'engine_index' as the tag and 'bitMaskAlarm:0' through 'bitMaskAlarm:7' as triggers.
- Right Pane (Properties):** Shows the configuration properties for the selected alarm. A blue circle with the number '3' highlights this pane. The properties include Name, Groups, Enable, Ack, Reset, Buffer, Trigger, Tag, Remote Enable, Remote Ack, Ack Notify, User Action, Description, Color, AckBlink, Severity, Events, Custom Field 1, and Custom Field 2.

Name	Groups	Enable	Ack	Trigger	Tag	Action	Description
Alarm1		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:0	engine_index		
Alarm2		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:1	engine_index		
Alarm3		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:2	engine_index		
Alarm4		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:3	engine_index		
Alarm5		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:4	engine_index		
Alarm6		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:5	engine_index		
Alarm7		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:6	engine_index		
Alarm8		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:7	engine_index		

Property	Value
Name	<Multiple selection>
Groups	
Enable	true
Ack	false
Reset	false
Buffer	AlarmBuffer 1
Trigger	<Multiple selection>
Tag	engine_index
Remote Enable	none
Remote Ack	none
Ack Notify	none
User Action	
Description	
Color	...
AckBlink	false
Severity	1-low
Events	76,76,1,1
Custom Field 1	
Custom Field 2	

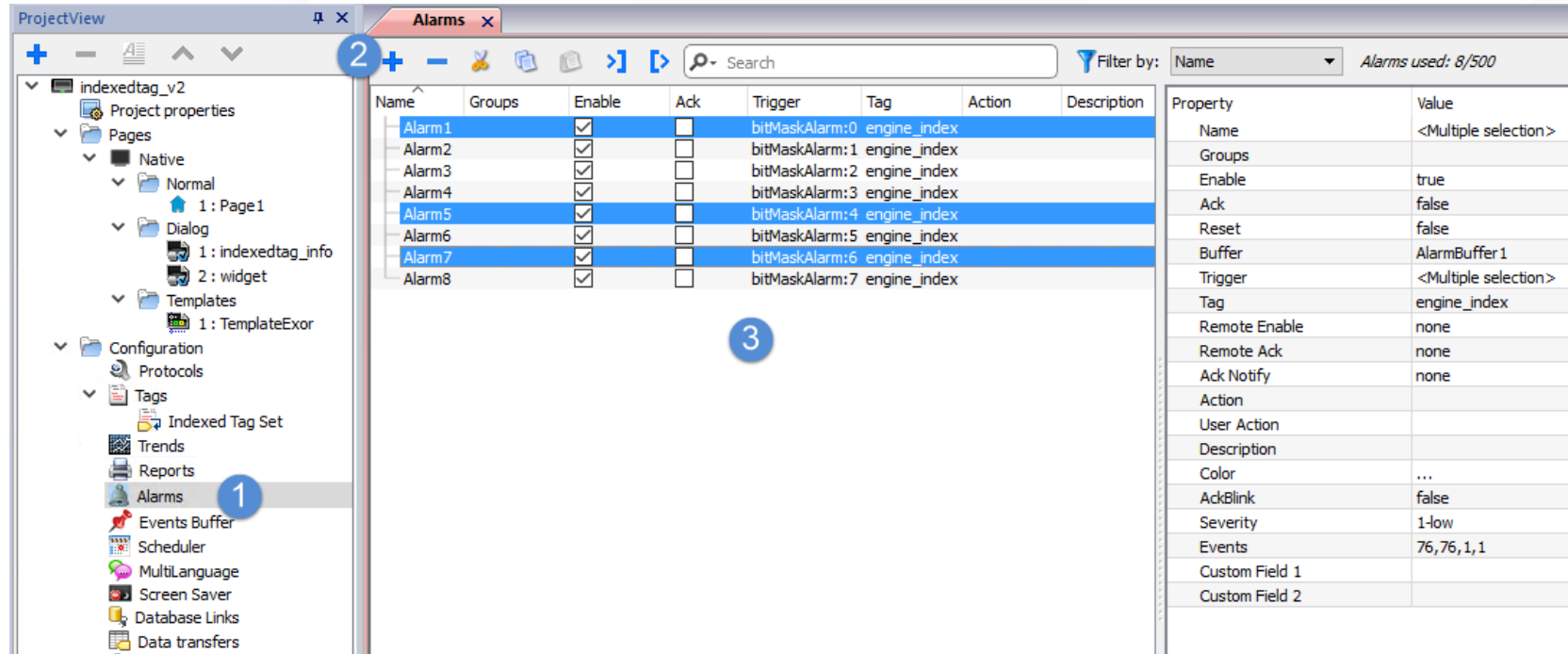
Working with Alarms

Alarms can be **Enabled**

Use **Trigger** to choose method

Select **Tag** to monitor as Alarm source

Use **Remote** features to synchronize alarm status between many HMIs



Working with Alarms

Select the **Action** to perform when alarm is triggered

Configure a **User Action** to perform by clicking on an active alarm

Set a **Description** to be viewed in Alarm widgets

The screenshot shows the EXOR software interface with the 'Alarms' configuration window open. The interface is divided into three main sections:

- ProjectView (Left):** A tree view showing the project structure. The 'Alarms' folder under 'Tags' is highlighted with a blue circle and the number '1'.
- Alarms Table (Center):** A table with columns: Name, Groups, Enable, Ack, Trigger, Tag, Action, and Description. It lists eight alarms (Alarm1 to Alarm8) with their respective configurations. The 'Action' column is highlighted with a blue circle and the number '2'.
- Properties Panel (Right):** A table showing the configuration for the selected alarm. The 'User Action' property is highlighted with a blue circle and the number '3'.

Name	Groups	Enable	Ack	Trigger	Tag	Action	Description
Alarm1		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:0	engine_index		
Alarm2		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:1	engine_index		
Alarm3		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:2	engine_index		
Alarm4		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:3	engine_index		
Alarm5		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:4	engine_index		
Alarm6		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:5	engine_index		
Alarm7		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:6	engine_index		
Alarm8		<input checked="" type="checkbox"/>	<input type="checkbox"/>	bitMaskAlarm:7	engine_index		

Property	Value
Name	<Multiple selection>
Groups	
Enable	true
Ack	false
Reset	false
Buffer	AlarmBuffer 1
Trigger	<Multiple selection>
Tag	engine_index
Remote Enable	none
Remote Ack	none
Ack Notify	none
Action	
User Action	
Description	
Color	...
AckBlink	false
Severity	1-low
Events	76,76,1,1
Custom Field 1	
Custom Field 2	

Working with Alarms

Choose alarm **Color** based on Alarm state, or put **Blink** on Ack

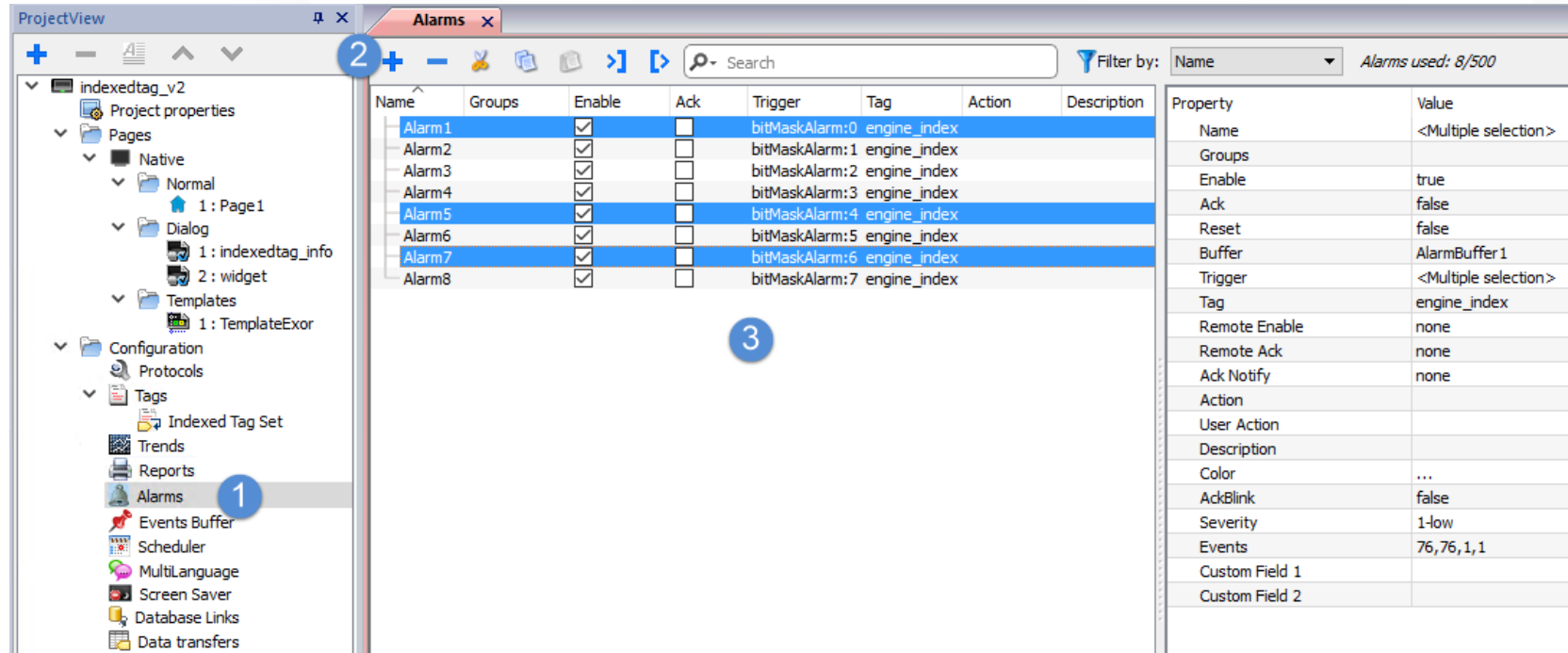
Organize alarms per **Severity**

Manage **Events**:

- If include alarm in Active widget or Historical widget

- When execute action (on alarm activation or de-activation)

- Select alarm information to be printed in text reports



Configure Trigger

bitMaskAlarm

Alarm triggered on status of bit

Triggers	Properties
<ul style="list-style-type: none"> limitAlarm bitMaskAlarm deviationAlarm valueAlarm 	<ul style="list-style-type: none"> bitMaskAlarm BitPositions: 0

limitAlarm

Alarm triggered when Tag value is out of "Min – Max" interval

Triggers	Properties
<ul style="list-style-type: none"> limitAlarm bitMaskAlarm deviationAlarm valueAlarm 	<ul style="list-style-type: none"> limitAlarm Min: 0 + Max: 1000 +

deviationAlarm

Alarm triggered when Tag value is out of range defined by Percentual Deviation over SetPoint

Triggers	Properties
<ul style="list-style-type: none"> limitAlarm bitMaskAlarm deviationAlarm valueAlarm 	<ul style="list-style-type: none"> deviationAlarm Deviation %: 50.0 + SetPoint: 20.0 +

valueAlarm

Alarm triggered on specific Tag value

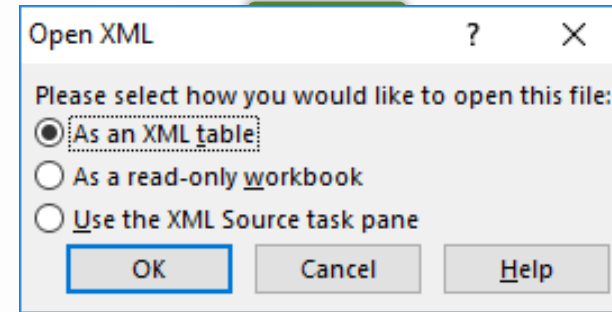
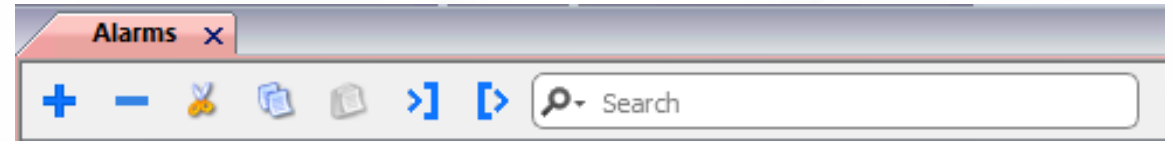
Triggers	Properties
<ul style="list-style-type: none"> limitAlarm bitMaskAlarm deviationAlarm valueAlarm 	<ul style="list-style-type: none"> valueAlarm Value: 0 +

Import/Export Alarms

Export made into ".xml" file format

Easy to open/edit with Microsoft Excel

Import back into Studio

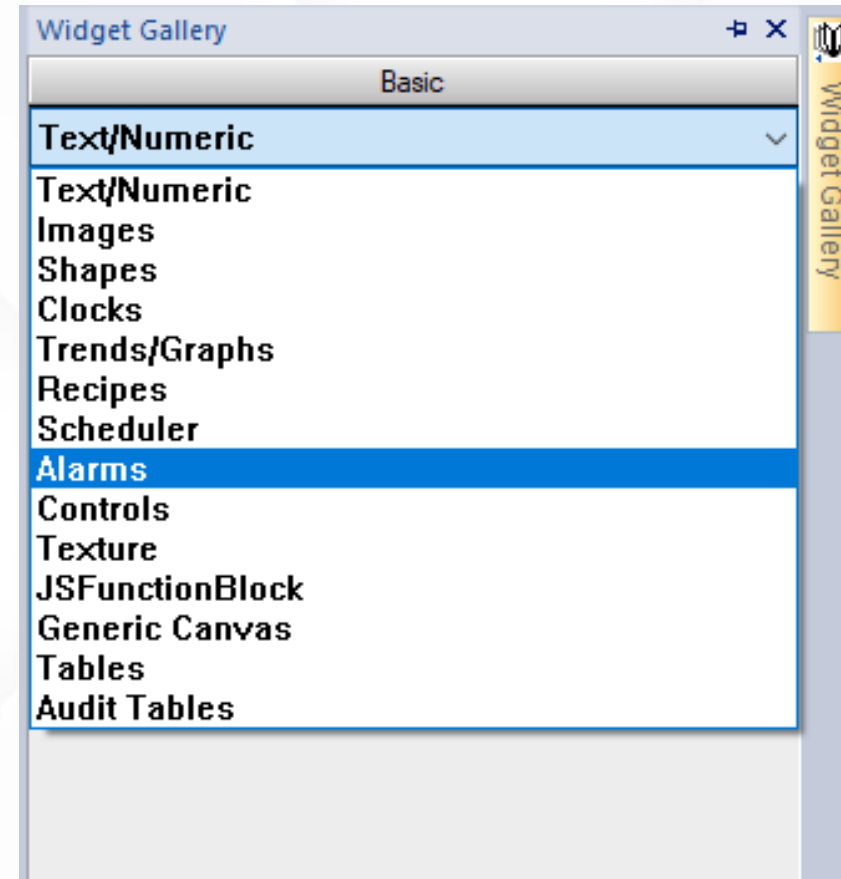


	A	B	C	D	E	F	G	H	I
1	eventBuffer	logToEventArchive	eventType	subType	storeAlarmInfo	name	source	alarmType	lowLim
2	n/a	TRUE	0	0	FALSE	n/a	n/a	n/a	
3	AlarmBuffer1	TRUE	14	1	TRUE	Alarm1	Tag1	bitMaskAlarm	

Alarms Widgets

Active Alarms
displays current active alarms list

Alarms History
displays contents of Alarm Events Buffer



Active Alarms

Different level of details

Lite: displays only alarm list

Acknowledge: allow Ack operations

Full: access to all Alarms features

Active Alarms - Lite

Name	Value	Time	Description
------	-------	------	-------------

Active Alarms - Acknowledge

Select	Name	State	Value	T
--------	------	-------	-------	---

Check/Uncheck All Ack

Active Alarms

Select	Name	State	Value	T
--------	------	-------	-------	---

Filter : Hide Not Triggered Check/Uncheck All Ack Reset Save

Alarms History

Displays content of Alarm Events Buffer




Contains widgets for navigation inside buffer

Reports time period displayed

Alarms History

From : 07/30/18 - 12:46:28 Duration : 1 Min Refresh

To : 07/30/18 - 12:46:28

Name	State	Value	Time
  			

Backward Forward

Alarm Widgets

- Both alarm widgets can be customized
 - enlarging/hiding Columns
 - hiding Buttons (ACK, Reset, Save...)
 - defining a second Filter (Active Alarm only)

Active Alarms

Select	Name	State	Value	T

Filter : Hide Not Triggered Check/Uncheck All Ack Reset Save

Properties

Alarms List : ActvAlrm

Columns	
Sorting	false
Sort Order	Descending
Sort Column	Severity
Text	
Filter	
Filter Column 1	State
Filter 1	Hide Not Triggered
DataLink	itemData:ActvAlrm.fltrcmb R
Access Type	R/W
Filter Column 2	Select
Filter 2	a
Table Antialiasing	
Header	

Table Alarm Widgets

Based on Table widget

More customizable in graphic

More efficient in performances

Active Alarms Table

Select	Name	State	Value	Time
<input type="checkbox"/>	Label	Label	Label	07/30/18 - 12:51:53

Hide Not Triggered
Toggle selection
Ack
Reset
Save

Alarms History Table

From: 07/30/18 - 12:53:22 To: 07/30/18 - 12:53:22 Duration: 2 Hours Refresh

Column Filter: Name 99999

Timestamp	Name	State	Value	Description
Label	Label	Label	Label	Label

Backward
Forward

Scheduler

Scheduler

Program the execution of specific actions
at repeated intervals
on a time basis

Feature to execute action on Sunrise or Sunset

Parameters of the schedule are
defined into Scheduler Editor

Can be changed at run time with Scheduler widget

Working with scheduler

Double click on "Scheduler" to open the Scheduler editor

Click "Add" to add a new schedule

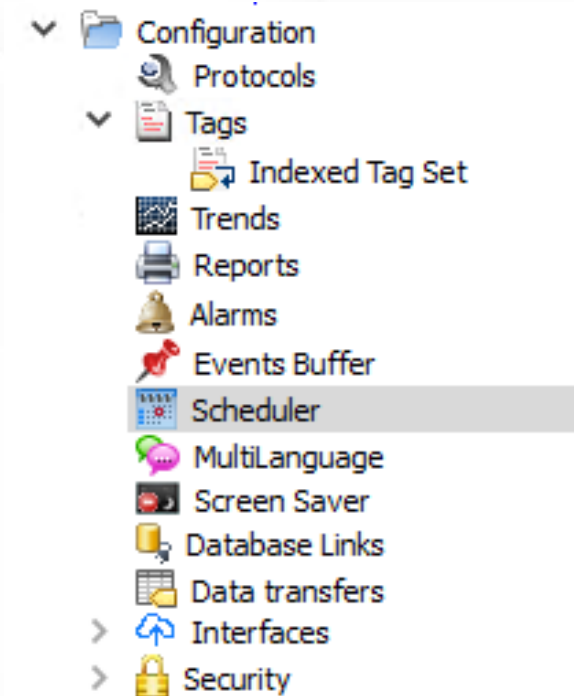
Give a name

Choose the type:

Recurring

High Resolution

Select the schedule properties



ID	Name	Type	Schedule	Action	Priority
1	StepTag	HighResolution	Every 500 msec	StepTag	Medium
2	ToggleBit	Recurring	Every, Time, 00:00:05	ToggleBit	Medium
3	ToggleBit_with_condition	Recurring	Every, Time, 00:00:05	ToggleBit	Medium
4	ShowDialog	Recurring	Weekly, Time, <...>, 14:00	ShowDialog	Medium

Working with scheduler

Recurring scheduler has options to set

Type: allow to define recurrence of schedule

Mode: you can base recurrence on time, sunrise or sunset

Condition: boolean tag to activate schedule

Actions: action to perform

Enable: enable/disable (can be changed on HMI)

On Startup: execute also at startup of application

ToggleBit Properties

Type: Date:

Mode: Time:

Condition: Location:

Actions:

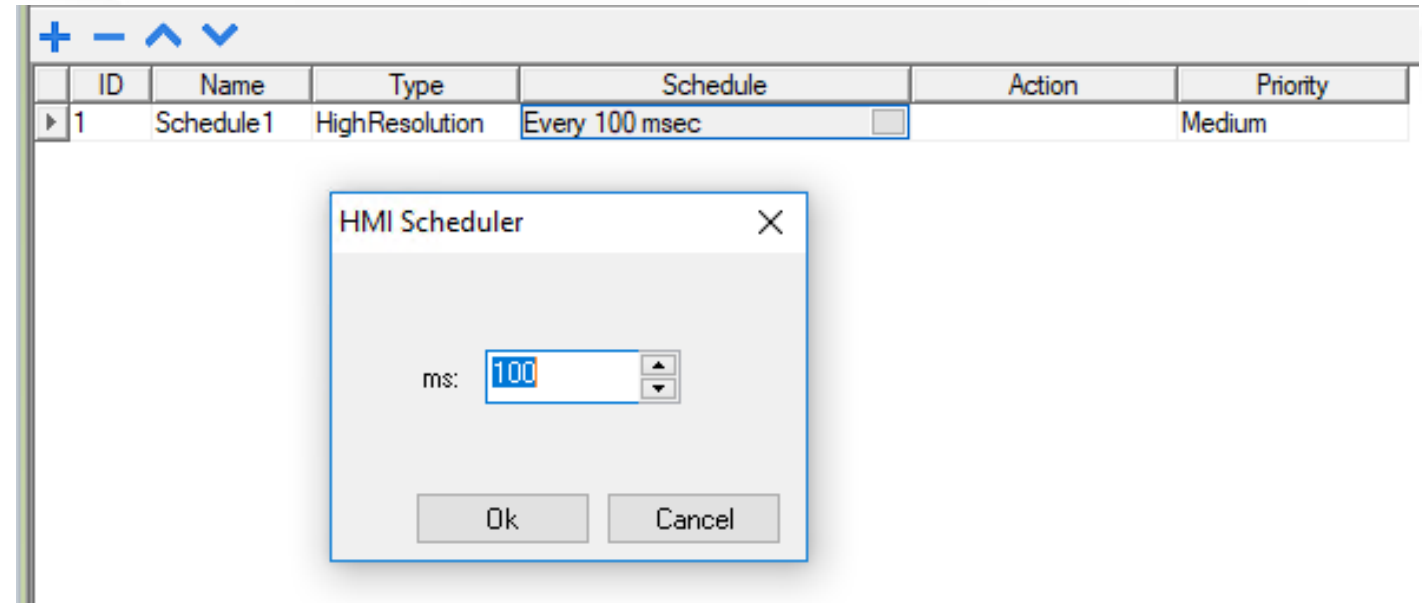
Mon
 Tues
 Wed
 Thurs
 Fri
 Sat
 Sun

On startup
 Enable schedule

Working with scheduler

High Resolution scheduler

Executed in cycle only,
minimum resolution 100msec



Scheduler widget

NEW
in 4.0

Change occurrence runtime

Based on table widget

Graphically customizable

Scheduler

Enable	Name	Type	Mode	Time	Occurrence	Condition	Edit
<input checked="" type="checkbox"/>	Schedule1	Every	Time	00:01:01		None	

Select Interval

01:03:05

▲
▼

Cancel
Prev
Next

Multilanguage

Multilanguage

Support for multiple languages

Add languages to the project and defining writing system

- Each string can have its own font

- Depending on selected language

- Studio shows fonts compatible with selected writing system

- Required fonts are downloaded with project or provided on external memory

Export/Import of language strings

Changing language at run time

Working with Multilanguage

Choose default language and set a custom name

Set a language code ISO 639 as language code identifier

Supports any writing system (fonts are filtered by specific selection)

Font management: displays number of fonts used

Possibility to use a font only on demand (saved on external device), due big size

The screenshot shows a software window titled "MultiLanguage" with a tab labeled "Text". Inside, there are "Add" and "Delete" buttons on the left, and "Save Font" and "Default" buttons on the right. A table below lists two language configurations:

	Language Name	Language Code	Writing system	Default Font	Fonts	Size	Storage
1	<Italian>	it	Any	Roboto	3	2.07 Mb	<input type="checkbox"/> Removable
2	English	en	Any	Tahoma	1	1.72 Mb	<input type="checkbox"/> Removable

Working with Multilanguage

Text in different languages can be added

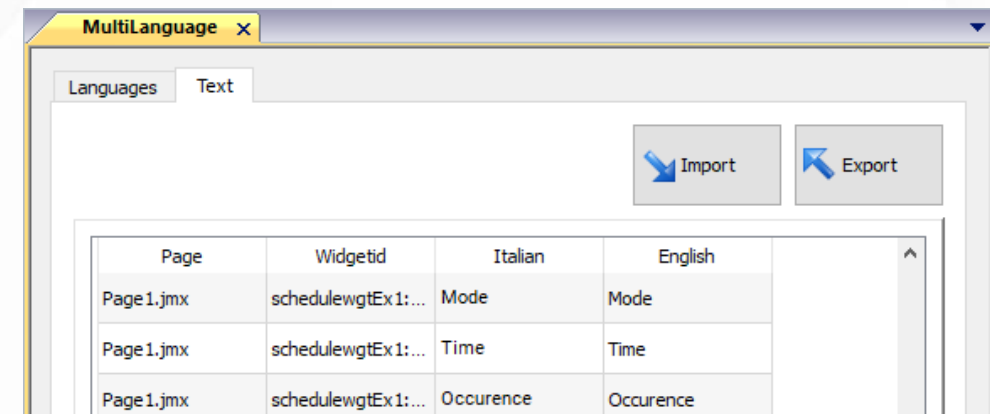
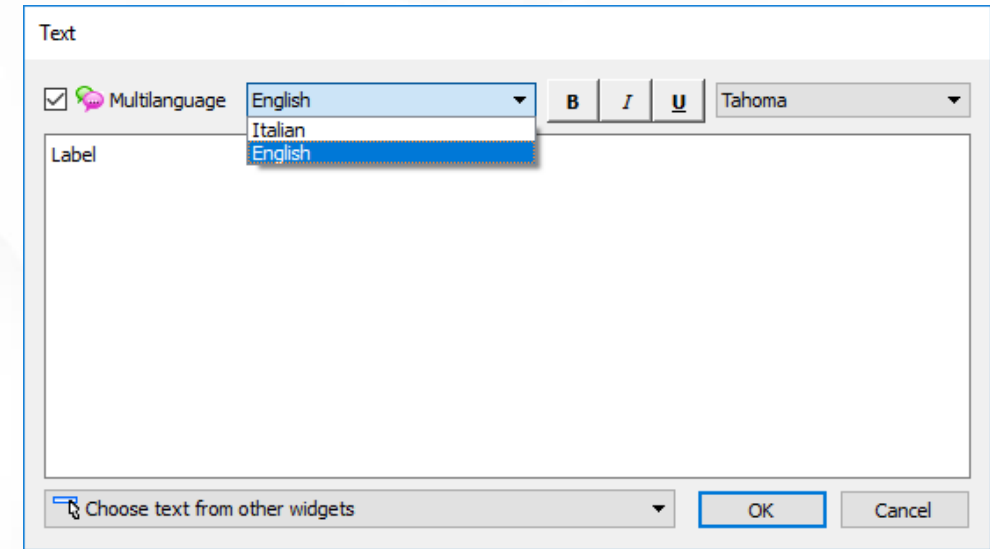
Typing text directly

Using "Text Table" in Multilanguage

Export/Import

Generates Unicode CSV

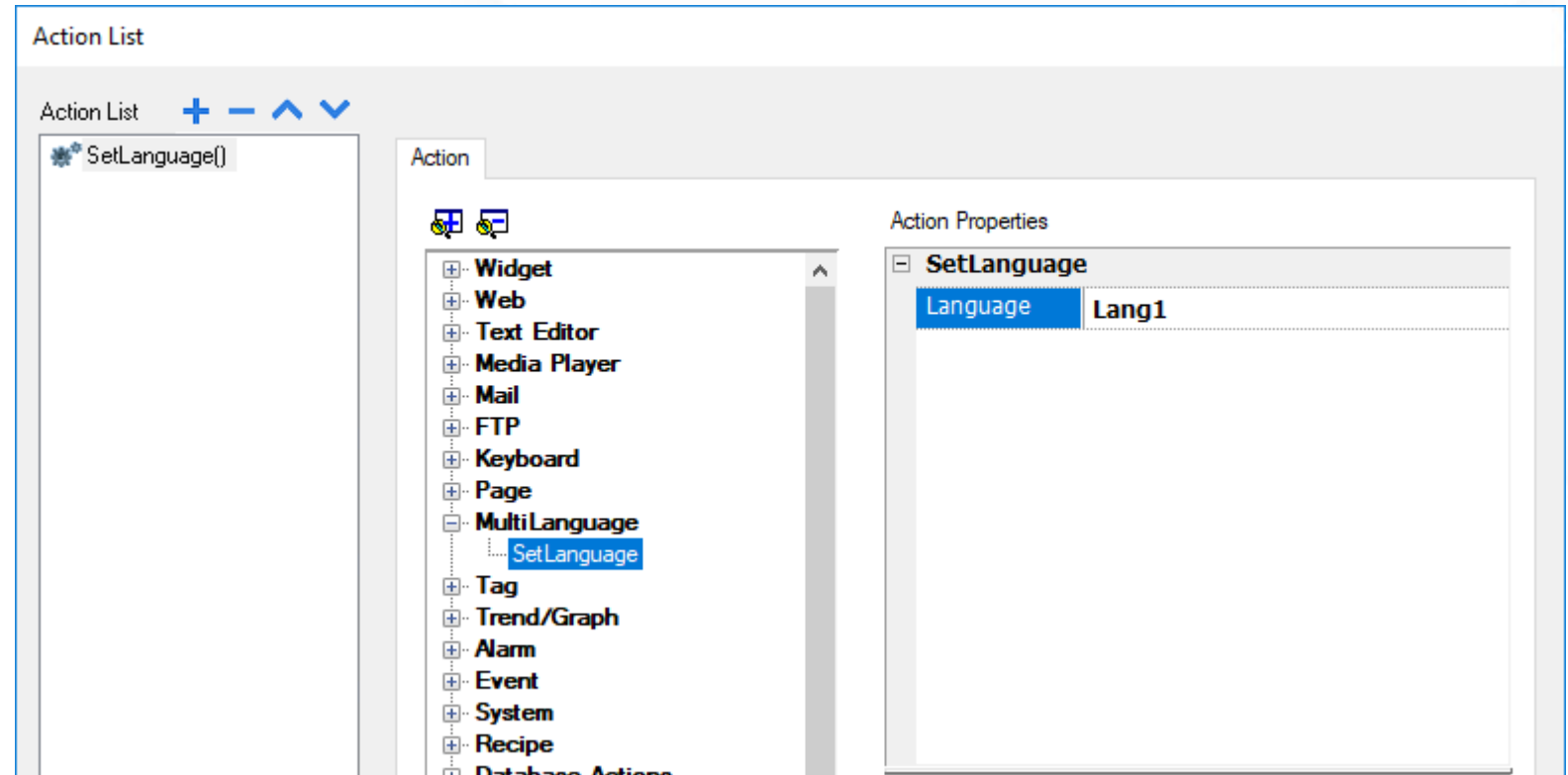
Can be edited with Microsoft Excel,
save as "Unicode Text (*.txt)"



Working with Multilanguage

Action to set specific language

Retentive setting



Screen Saver

Screen Saver

Screensaver allows to:

Create slideshows to be activated as screen savers

Images for slideshows can be resident on removable memory

Easily configure actions or scripts to be executed upon screensaver activation and deactivation

Screen Saver

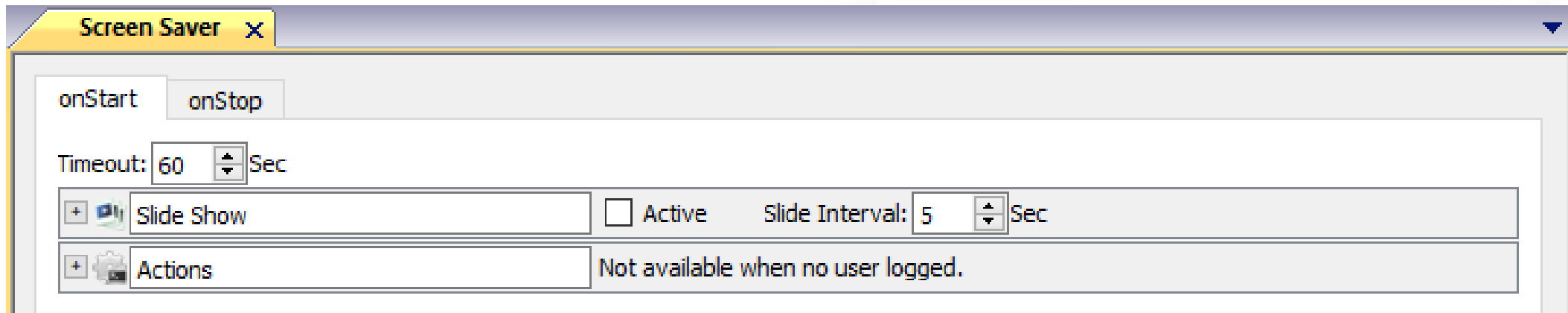
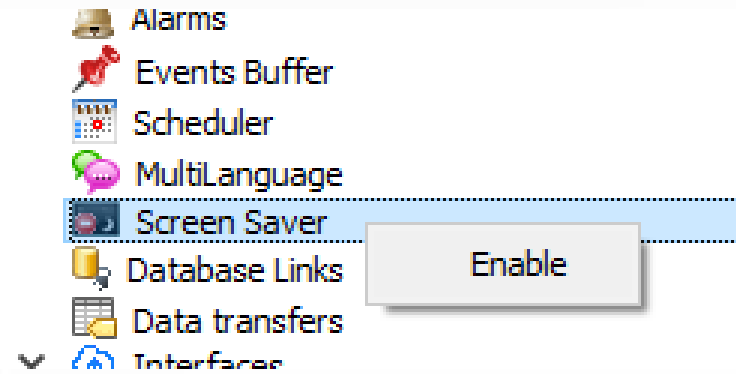
Screen Saver must be enabled

OnStart and OnStop event can be defined:

- Images Slide Show

- Actions

- Javascript



Data Transfers

Data Transfers

HMI becomes a data gateway
and an element for a better system integration

Move data between different controllers
in a fully programmable way

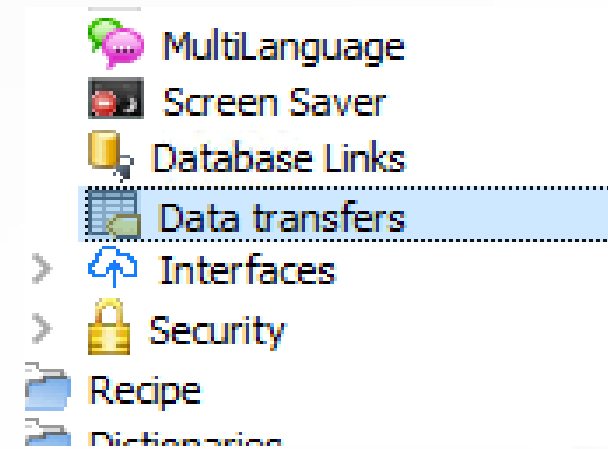
Bi-directional transfers can be programmed

Data Transfers

Data transfers can be done:

On update: when the value of the source tag changes

On trigger: when tag defined as Trigger changes.
Trigger limits can be applied if necessary



Data transfer									
	TAG A	TAG B	Direction	Update method	Trigger	Low limit	High limit	Enable	on Startup
1	Tag1	Tag2	A->B	On update		0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	Tag3	Tag4	B->A	On trigger	Tag5	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Security and User Management

Security and User Management

User management

- Groups with set of authorizations
- Users belonging to Groups

Users password options:

- Common settings
- Change initial password
- Auto Logout timer
- Option for allowed passwords



Security and User Management

User editing at runtime

Action Login/Logout/Switch users

Each user has Home Page

Option to Switch showing last visited page

Compliant to FDA 21 CFR Part 11



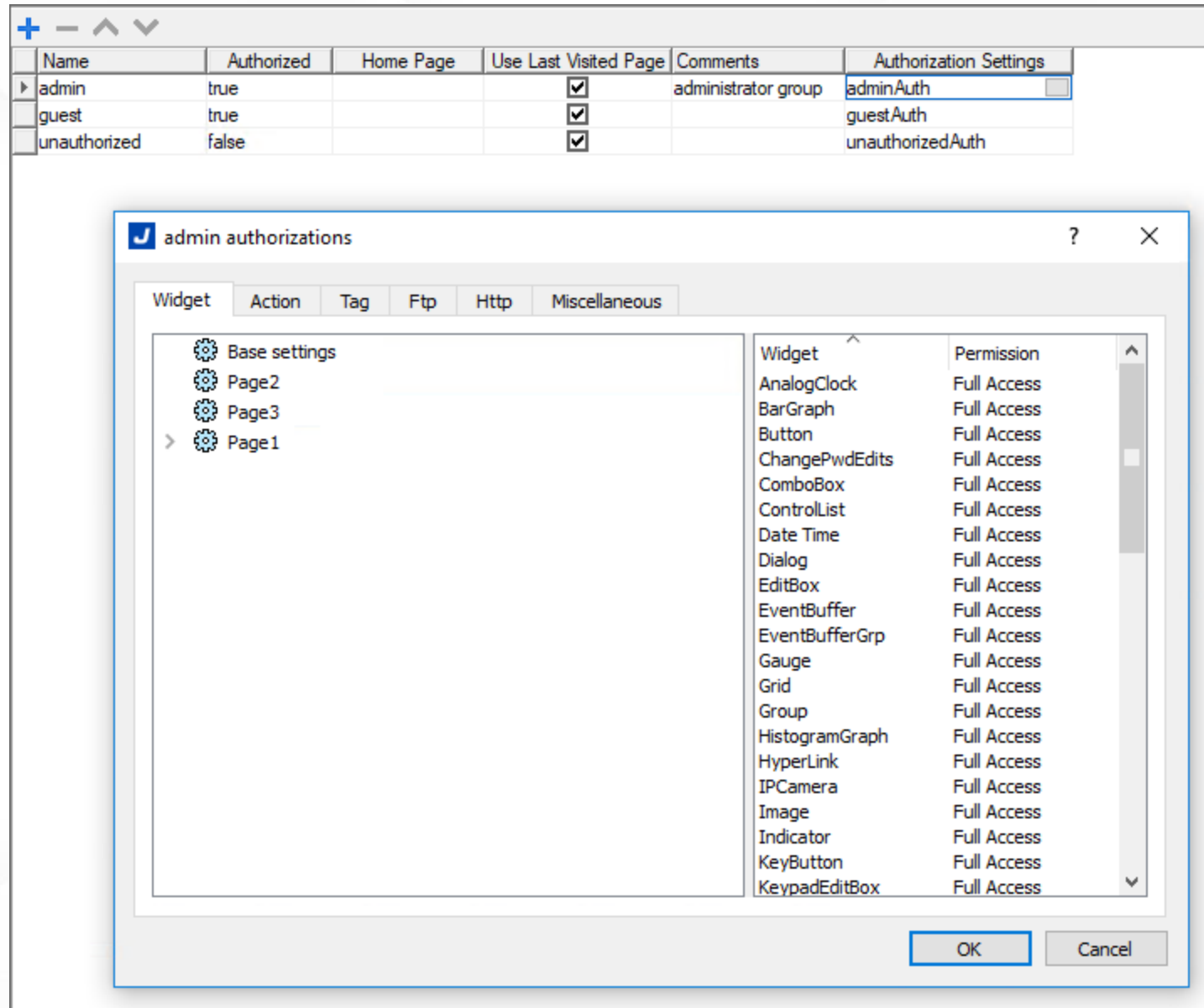
User Group Authorizations

Define groups

Define per each group the "base settings" authorizations

By widget:
Hide, Read Only, Full Access

By Action:
Allowed, Not Allowed



User Group Authorizations

NEW
in 4.0

Tag permissions
allow read/write permissions

Enable FTP server

Adjust HTTP permissions
by IP address

Name	Authorized	Home Page	Use Last Visited Page	Comments	Authorization Settings
admin	true		<input checked="" type="checkbox"/>	administrator group	adminAuth
guest	true		<input checked="" type="checkbox"/>		guestAuth
unauthorized	false		<input checked="" type="checkbox"/>		unauthorizedAuth

admin authorizations

Widget Action Tag Ftp Http Miscellaneous

Widget	Permission
Base settings	
Page2	
Page3	
Page1	
AnalogClock	Full Access
BarGraph	Full Access
Button	Full Access
ChangePwdEdits	Full Access
ComboBox	Full Access
ControlList	Full Access
Date Time	Full Access
Dialog	Full Access
EditBox	Full Access
EventBuffer	Full Access
EventBufferGrp	Full Access
Gauge	Full Access
Grid	Full Access
Group	Full Access
HistogramGraph	Full Access
HyperLink	Full Access
IPCamera	Full Access
Image	Full Access
Indicator	Full Access
KeyButton	Full Access
KeypadEditBox	Full Access

OK Cancel

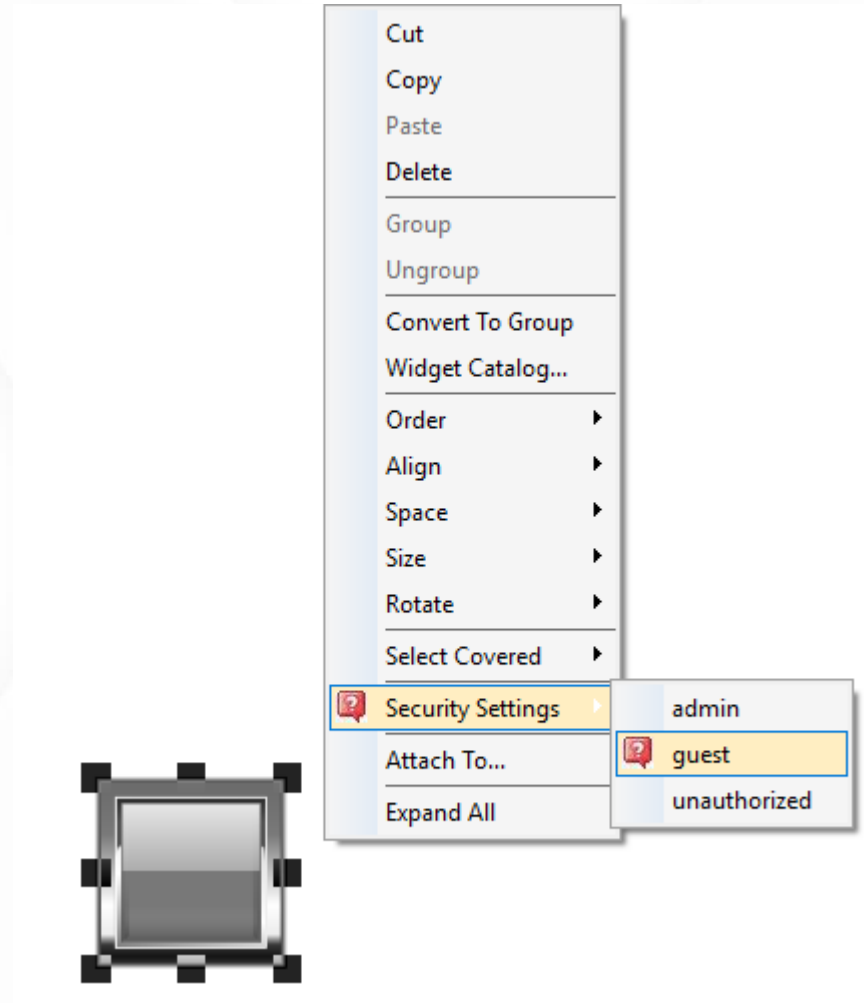
Widget Authorizations

Permissions can be set on single widget

Right Click

Security Settings

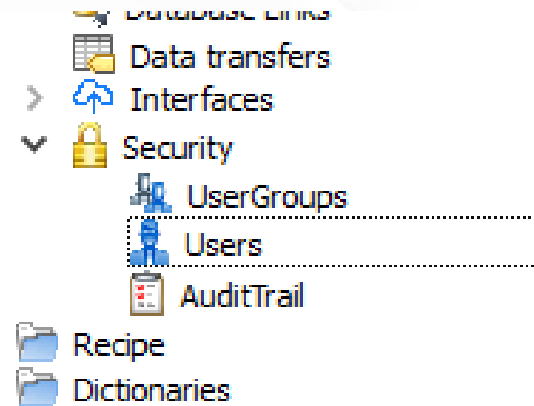
Select group



Create Users

At design time, from Users

Common Parameters



The screenshot shows the 'Users' management window with a table of users and an open 'Common Parameters' dialog box. A blue arrow points from the gear icon in the toolbar to the dialog box.

Name	Default User	Group
admin	<input checked="" type="checkbox"/>	admin
user1	<input type="checkbox"/>	guest

password	Logoff Time (minutes)
	0
	0

Common Parameters

Parameters common to all users

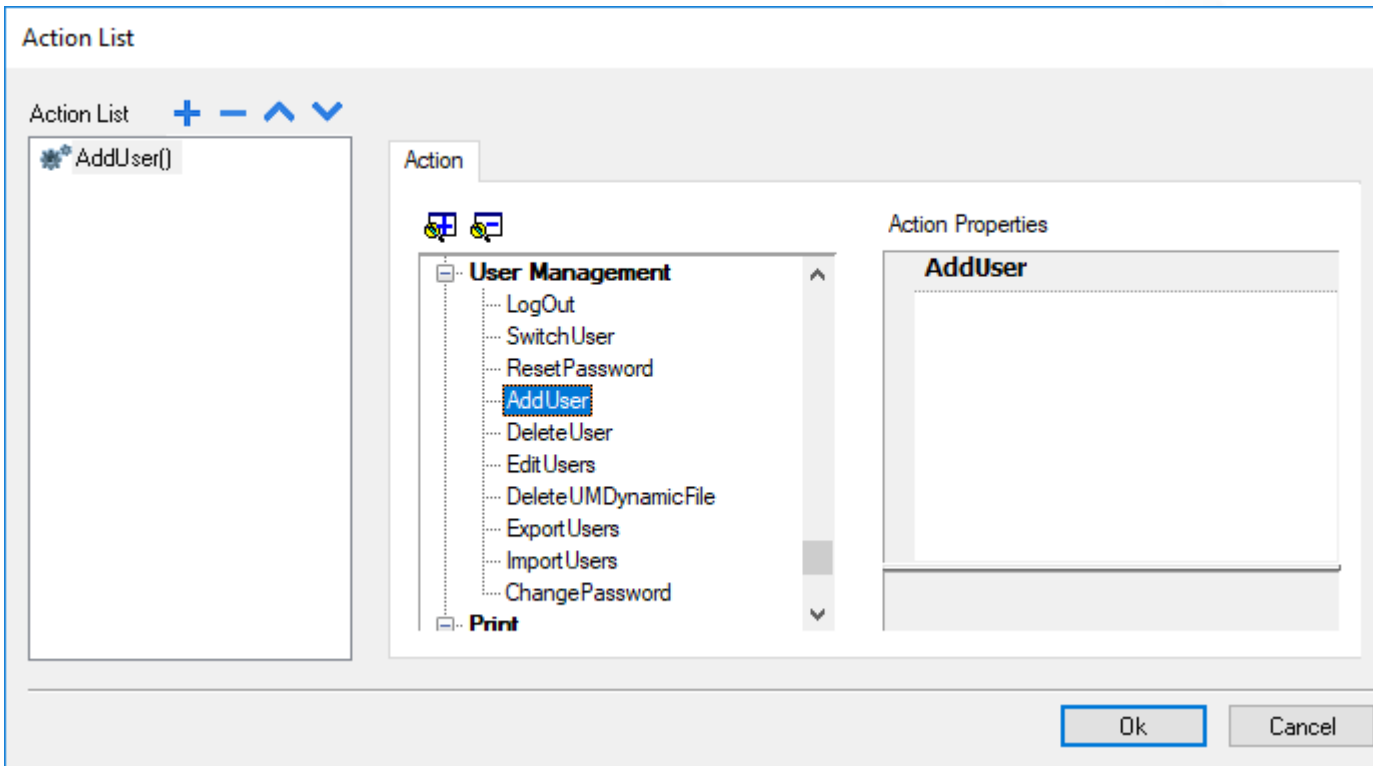
- Force to change initial password
- Force log off time (min)
- Minimum Password Length (Char)
- Password must contain special characters
- Password must contain numbers
- The last three passwords cannot be reuse
- Number of weeks before force a password change
- Show warnings before password expired (day)

Possibility to set specific user as "Inactive"



Create Users

At runtime, using dedicated action



User name:

Password: Show password

Group:

Comments:

User must change his initial password

Inactivity logoff time (Min)

Recipes

Recipes

- Configuring recipes

 - Recipe, Element, Set

- Storing recipes data

 - Recipe data files

- Extracting data

 - FTP

 - Backup and Restore to USB

- Handling recipes at run time

 - Menu selection

 - Download/Upload

 - Add/delete recipe sets

Working with Recipes

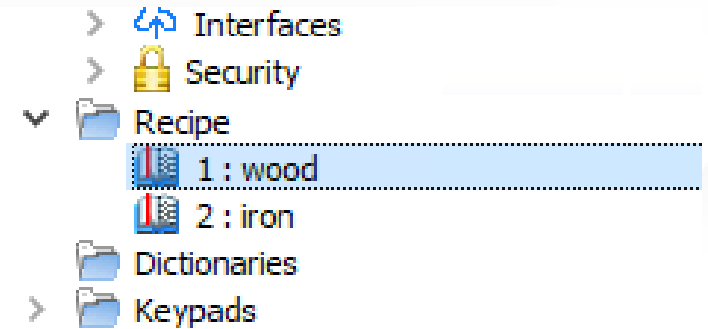
Right click on "Recipes" and select "Insert Recipe"

Double click on Recipe name to open the recipe editor

Configure

Elements (rows): associate to a Tag

Sets (columns): predefined values for each Tag



Recipe x

index	Element Name	Tag	box1	box2	box3	box4	box5
0	Element1	lenght_w	10	11	12	8	5
1	Element2	width_w	20	22	24	16	10
2	Element3	height_w	30	33	36	24	15

Properties

Recipe : _RecipeMgr

Recipe Name	wood
Number of sets	5
Set 0	box1
Set 1	box2
Set 2	box3
Set 3	box4
Set 4	box5

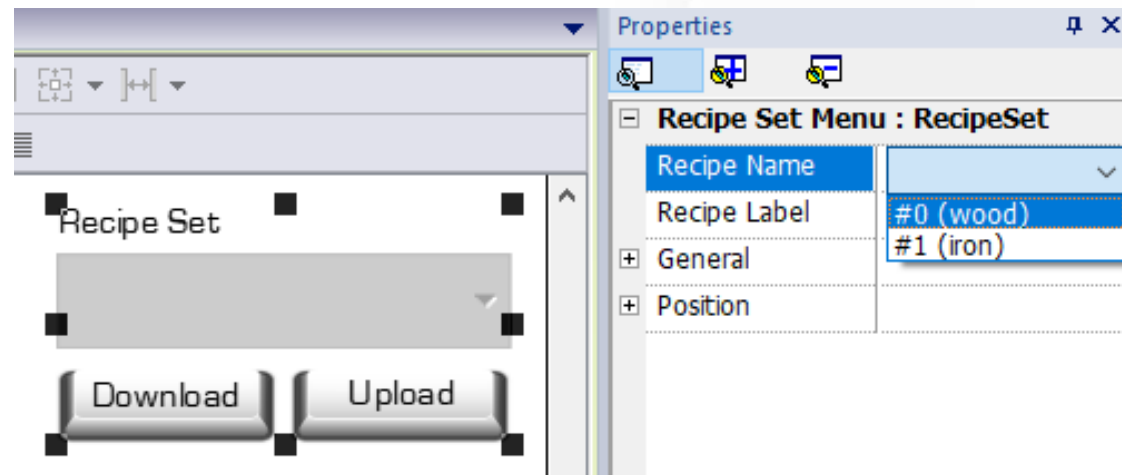
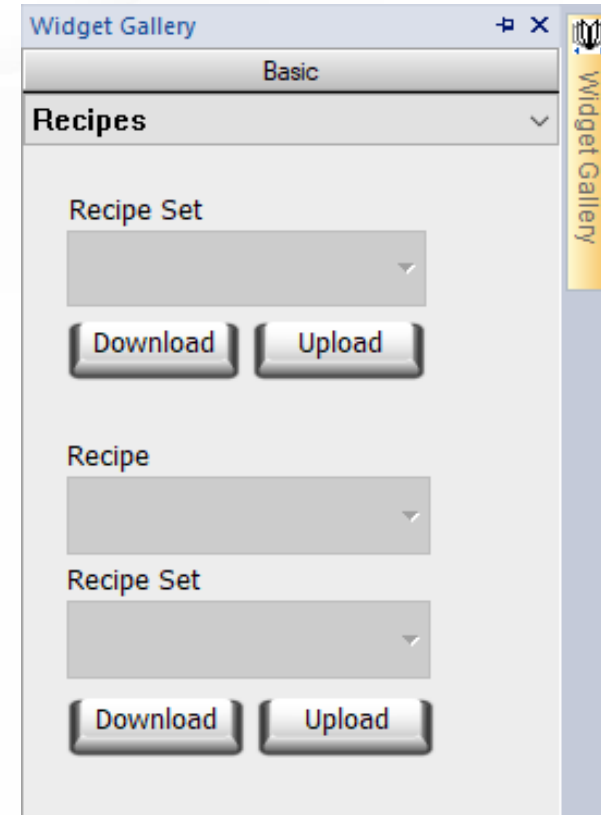
Working with Recipes

Locate Recipe widgets in gallery
 Basic category > Recipes

Two different widgets:

Recipe Set: user can select only Set of a specific Recipe

Recipe Menu: user can select Recipe and Set on HMI



Add/Remove Recipe Sets

Number of recipe sets can be changed at runtime (add/remove)

Recipe editor allow to move recipe sets

index	Element Name	Tag	box1	box2	box3	box4	box5
0	Element1	lenght_w	10	11	12	8	5
1	Element2	width_w	20	22	24	16	10
2	Element3	height_w	30	33	36	24	15

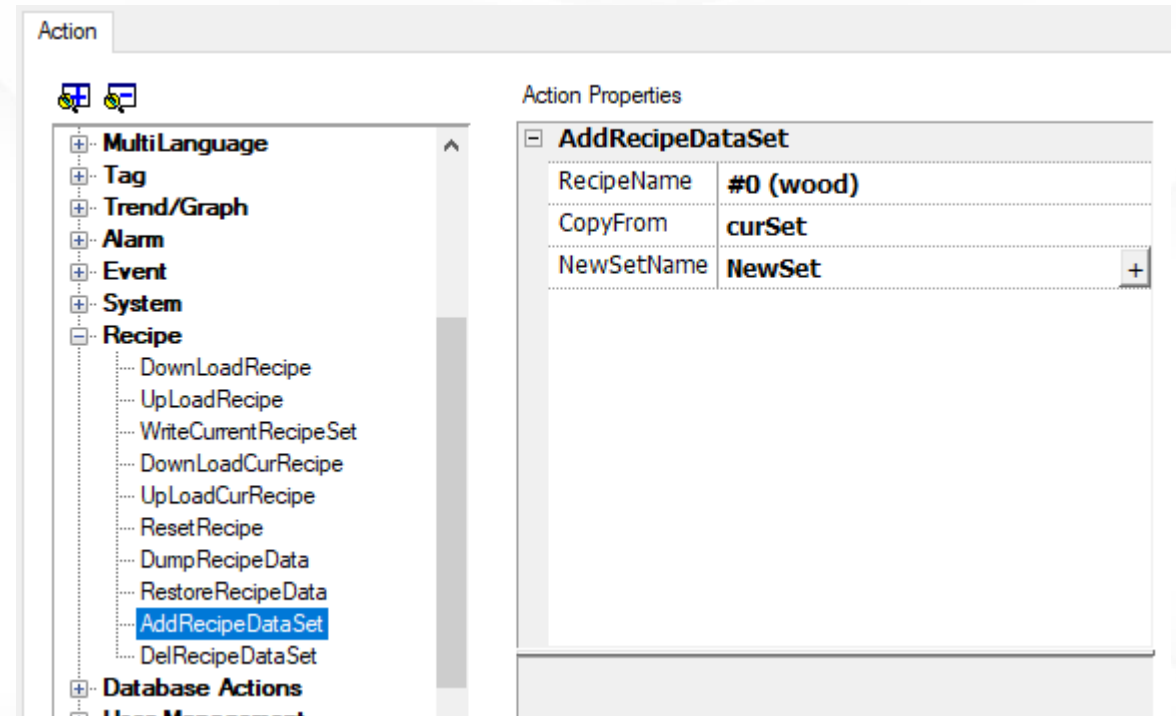
Add/Remove Recipe Sets

Actions to Add and Delete a Set

RecipeName: into which recipe add/delete the set

CopyFrom (opt): from which set copy values

NewSetName: default name for the new set



Action to Dump and Restore a Recipe

Restore by Replace, Match, Match and Add

Recipe data on widgets

Static link



Dynamic link

Data on page changes according to Recipe widget current selection



End of JMobile Training Day 1

Thanks for your attention
Technical Support Team