BeiJer

E L E C T R O N I C S

iX Developer

Reference Manual

English

Reference manual for iX Developer

Foreword

The iX Developer software is used to configure Operator panels and PC operated control applications, including applications for IPCs (Industrial PCs).

The iX Developer makes it easy to create logical, flexible and effective HMI applications that provide the right information on the right occasion to operators and to other systems.

This manual describes the configuration software in detail.

Please see the iX Developer User's Guide (MAEN832x) for function-based descriptions.

The manual assumes that the most recent versions of the system program (image) and iX Developer are used.

For specific details of a connected controller refer to the help file for the controller driver. The function of a project application in an operator panel is not affected by the choice of controller.

The information in this manual is also available by pressing F1 while using iX Developer.

The present revision of this manual is valid for version 2.0 of iX Developer.

Order no: MAEN831G

Contents

1 The Configuration Tool

1.1 Introduction

The iX Developer software is used to configure Operator panels and PC operated control applications, including applications for IPCs (Industrial PCs) from Beijer Electronics.

iX Developer contains all basic functions needed in an application. The functions are tested and developed with customer needs and preferences in focus.

Pre-defined objects in iX Developer can be used to create complete process images, providing an overview of a complex application. You can customize the pre-defined objects or create objects of your own.

Communication drivers for a large number of controllers and automation equipment are available.

The help file assumes that the most recent versions of the system program (image) and iX Developer are used.

1.1.1 Controller

Operator panels can be connected to many types of automation equipment, such as PLCs, servos, and drives. Further on, the expression *controller* is used as a general term for the connected equipment.

Related information
Controller

1.1.2 Tags

Data values in a controller are referred to as *tags*.

Tags may also belong to the system or be internal. A tag has a symbolic name and can be of different data types.

Objects connected to tags can change values in the controller, and tag values can be reflected by changing object appearance in various ways. Objects in a screen will remain static until connected to a tag.

Related information
Tags

1.1.3 System Requirements and Limitations

iX Developer

Parameter	Recommendation
RAM	2 GB
Processor	2 GHz or higher
Operating system	Microsoft Windows XP SP3
	Microsoft Windows 7
	Microsoft Windows Vista
Graphic card	Pixel shader 3 or higher ensures full graphic acceleration

Updates

Software, drivers and protocols may have been updated since the USB stick was produced. Therefore, it is recommended that you use the built-in update function in iX Developer before creating a project.

Related information
Update Software
Updating Drivers

iX Runtime

Parameter	Recommendation
RAM	1 GB
Processor	1.3 GHz or higher
Operating system	Microsoft Windows XP SP3
	Microsoft Windows 7
	Microsoft Windows Vista
Graphic card	Pixel shader 3 or higher ensures full graphic acceleration

Special Requirements for Some Objects

For some objects to be included in the iX Developer project, specific software versions are required. Simulation of the project on the development PC may also be limited for some targets.

Object	Minimum requirement	Simulation on PC target	Simulation on panel target
Media Player	Microsoft Media Player version 10	Supported	Not supported
PDF Viewer	Acrobat Reader version 9	Supported	Not supported
Web Browser	Microsoft Internet Explorer 7	Supported	Not supported

Limitations for Keyboard Operated Panels

Some functions and objects are not supported for keyboard operator panels:

Limitations for keyboard operator panels
Key beep
Key repeat
PDF Viewer object
Web Browser object
Media Player object
Trend Viewer Legend tool
Database Viewer object
Filter dialog in Alarm Viewer object
Multiline in text box objects

1.1.4 Getting Started

iX Developer is installed on a development PC, where projects are developed, designed and compiled. The project is then run in an operator panel, IPC or PC to observe and control a controller (or a group of controllers).

Target

iX Developer projects can be targeted for

*   An operator panel from Beijer Electronics
*   An IPC (Industrial PC) from Beijer Electronics
*   A standard PC with Microsoft Windows XP Service Pack 3, Microsoft Windows Vista or Microsoft Windows 7

To enhance readability, only *operator panel* will be mentioned sometimes, when all different targets actually could be included.

The functions in iX Developer depend on the model of operator panel used.

Some differences exist between the targets, for example:

Item	Support on IPC or PC target	Support on operator panel target
Media Player	Supported	Not supported on Panel TxA
PDF Viewer	Supported	
Web Browser	Supported	
Navigation List Box	Supported	Not supported on Panel TA/TxA/TxB and Q-Term panel
Screen Carousel	Supported	Not supported on Panel TA/TxA/TxB and Q-Term panel
Dynamics	All dynamic settings are supported	Some dynamic settings are not supported on Panel TA/TxA/TxB, e.g. background color for button, trend viewer and meter, and min/max values for meter
Opacity	Supported	Not supported on Panel TA/TxA/TxB and Q-Term panel
Drop down shadow effect	Supported	Not supported on Panel TA/TxA/TxB and Q-Term panel
Rounded corners on rectangle	Supported	Not supported on Panel TA/TxA/TxB and Q-Term panel
Report	Supported	Reports containing charts and pictures can not be printed from an operator panel target.

License

A project for an Operator panel can be used without any restrictions imposed by licensing.

A limited number of controller tags are available for a standard PC project. The number of tags is controlled by a USB hardware dongle.

The industrial PCs (IPCs) from Beijer Electronics are configured with a fixed limit of tags, and do not require a USB dongle.

The number of used controller tags (including DEMO controller tags) is shown in the lower right of the desktop. The figures will turn red if the number of used tags becomes larger than the number of available tags.

There are no license restrictions for internal tags.

Related information
Internal Tags
DEMO Controller

Product Registration

The first time iX Developer is started, a registration dialog is displayed, in order to supply you with a registration key. Entering the registration key provides unlimited access to all program functionality and software updates.

Alternatively, select to continue using a trial version of the software. You may evaluate iX Developer with full functionality for 30 days. When the evaluation period has expired, it will still be possible to use the software, but with reduced functionality.

Note:
If you have already used a Demo version of iX Developer for 30 days, you will not be issued another 30 days for evaluation.

Project Size

The project size is shown in the lower right of the desktop area when designing an operator panel project. The size was calculated at the latest validation.

1.1.5 Installation

The iX Developer software is supplied on a USB stick. When connecting the USB stick to the USB port, the installation will start automatically. Follow the instructions to install iX Developer. If the installation does not start, run the setup.exe file.

The installation creates an icon for iX Developer in the group named iX Developer.

Click on **Start/All Programs/iX Developer/ iX Developer** to start iX Developer.

1.1.6   Project

The top folder for a specific application designed with iX Developer is referred to as the project folder.

During runtime, project database files can be updated, for example with new recipes. This means that to completely reproduce a project that has been in operation, it may be necessary to combine source files with files retrieved from the operator panel.

1.1.7   File Structure

A project contains a set of files related to the functional and graphical design and a set of files related to the runtime operation of the project, where the latter is compiled from the design files.

Project Folder

When a new project is created, a folder with the project name is created as the top-level container, the **Project** folder. The files that define the functional and graphical design reside in the top level of the project folder. Other folders are created as a result of validation and build.

Symbols

Pictures that are used in projects are converted to .png files when the project is validated.

Pictures are resized to the largest static usage in any of the project screens, in order to save memory space in the panel. If a picture is enlarged in runtime using dynamics, the enlarged picture will have a lower effective resolution.

Project pictures are stored in the **Symbols** folder as a compressed folder named Symbols.zip.

Temp

The **Temp** folder contains intermediate build files from project compilation.

The **Temp** folder also includes the **Output** folder. The **Output** folder contains all files needed to run the project in the target. These files are copied to an operator panel when the **Transfer** command is used. For a PC target, the **Transfer to folder** command is used to copy the necessary files to a USB stick to the runtime PC via a network connection.

Moving Design Files

To move the project files needed for the design:

1. Create a new folder for the project design files.
2. Copy all single files (files not included in any folders) as well as all folders, except the **Temp** folder, in the project folder.
3. Paste the files in the new folder.

Moving Application Project

The runtime project can be downloaded to the operator panel using the transfer or export commands, but can also be moved manually:

1. Create the new folder for the project files.
2. Copy the **Output** folder.
3. Paste it in the new folder.

Database

The database used for iX Developer projects is of SQL Server Compact Edition type. The contents of the database can be managed with third-party database management tools. iX Developer includes a database viewer object that can be used to display database contents in runtime.

The database is initially transferred to the operator panel using the transfer command. The database will only be updated if there are changes made in the project that affects the database contents, for example:

- Changed or added recipes
- Changed or added data loggers
- Changes to alarm handling
- Changed or added trend viewers

A copy of the database can be made with the **Upload Database** command. Some of the databases can be exported individually in csv format using the **Database Export** action. If possible, it is recommended to use the database export method rather than the backup method.

Related information
Database Viewer Object
Download
Export
Upload Database
Database Export

2 Working with Projects

This section describes iX Developer and explains how to work with a project for an operator panel.

In iX Developer, ribbons tabs are used instead of menu commands. This reduces the number of steps needed to design complex components, and also supplies an attractive user interface.

Ribbon tabs are located in the top section of the tool window. Each ribbon tab holds one or several control groups. Each group contains a set of related controls. Controls are made to design screens, and to make settings for objects and controls in the project.

2.1 Creating a Project

A project can be created according to the following sections, but the work process can be adapted and rearranged if needed.

2.1.1 Connecting to a Controller

Establishing communication between an operator panel and controller is necessary to allow operator observation and control. The larger the project is, the more important it is to ensure that there is a suitable communication design.

There is a built-in DEMO controller in iX Developer. It can be used for test and simulation purposes.

Related information
Creating a New Project
Select Controller
Communication Design
DEMO Controller

2.1.2 Designing a Screen Set

It is important to organize the application well and to consider which functions are necessary. Start with an overall view and then work down to a detailed level.

A project contains a number of screens with objects that can exchange data with the controller. Screens can be arranged in hierarchies to achieve a structured application, or organized as sequence controls. When deciding which structure to use, consider how to best describe the process at hand and how to simplify procedures for the machine operator.

The complete application, or parts of it, can be tested in the development environment before downloading it to the operator panel.

Related information
Screens

2.1.3   Designing Additional Functions

Alarms

Alarms are used to make the operator aware of events that require immediate action. An alarm is set when a certain condition is met. An alarm condition is designed as a logical evaluation of a tag value. Alarms can be divided into groups to create an order of priority.

Related information
Alarm Management

Function Keys

Function keys can be used to perform actions and execute scripts. This allows operator control of data and screen functionality independent of which screen is active.

Related information
Function Keys

Multiple Languages

Translation of texts and system texts can be performed directly in the application, or via export to a text file to be translated in other software. The file is imported to the application after translation. The application language can be changed in runtime, for example based on a tag value.

Related information
Language Management

Security

Access to objects and actions in the project can be limited using security groups and user passwords.

Related information
Security Management

Text Library

With the text library function, text tables can be created, where values are linked to texts.

Related information
Text Library

Audit Trail

The Audit Trail function allows tracking of operator actions.

Related information
Audit Trail

Data Logger

Data can be logged and saved to file. Bit, 16-bit, 32-bit and Real (Float) values can be logged. Be aware that logging of data consumes system resources and memory.

Related information
Data Logger

Scheduler

Setting and resetting digital tags in relation to the real-time clock can be performed using a scheduler, in order to control events in the process at special calendar times.

Related information
Scheduler

Recipes

Recipes are used to set or save a predefined group of tags in one operation.

Values can be predefined or collected from the controller, and then saved to a recipe in the operator panel. The operator can download the recipe at any time to the controller, which will start working with the recipe values. Recipe handling makes it possible to reuse large parameter sets, to improve efficiency of time critical production where a change of products must be made quickly. Recipe files can be created in the development project or with the operator panel.

Related information
Recipe Management

Reports

The reports function allows adding excel report templates to the project.

Related information
Reports

2.2 Importing an Information Designer Project

Existing Information Designer projects can be imported to iX Developer. Information Designer is the previous software used to design projects for the EXTER operator panels.

The import function is accessed from the Welcome page, displayed when starting iX Developer.

1. Select the Information Designer project file.

2. Select a name and location for the imported project.

3. Click **Finish**.

The project will be imported to the corresponding Operator panel target. All screens will be imported and graphical objects in the Information Designer project will be converted to corresponding iX Developer objects. Some objects may not be supported.

Functions such as recipes and data logging work differently in iX Developer, and will have to be reconfigured in the imported project.

During the import, a log file of unsupported objects and functions is created. It can be saved to disk for future reference.

2.3 Importing an H-Designer/ADP Project

Existing H-Designer/ADP projects can be imported to iX Developer. H-Designer/ADP is the previous software used to design projects for the H-/PWS-series operator panels.

The following software versions are required:

Software	Version
ADP	6.50 build 184
iX Developer	2.0

2.3.1 Exporting the H-Designer/ADP Project

First, the H-Designer/ADP project has to be converted to an a2i file from within ADP by following the steps below:

1. Select **File/Export iX**.
2. Select a name and location for the export file, and click **OK**.

    A message box confirms that the export is finished, and after clicking **OK**, you are asked if you want to see the log file. The log file is saved to the same location as the project file.

    The log file declares successfully converted items as well as unsupported objects and functions.

2.3.2 Importing the a2i File

Then, the exported file is imported into iX Developer using the import function that is accessed from the Welcome page, displayed when starting iX Developer.

1. Select the a2i file.
2. Select a name and location for the imported project.
3. Click **Finish**.

The project will be imported to the corresponding Operator panel target. All screens will be imported and graphical objects in the H-Designer/ADP project will be converted to corresponding iX Developer objects. Some objects and functions may not be supported at all, and some objects and functions need reconfiguring in the imported project.

During the import, a log file of unsupported objects and functions is created. It can be saved to disk for future reference.

2.3.3 Limitations

All parts of the original H-Designer/ADP project will not be fully supported in iX Developer. For example, for objects that use individual controller registers for read and write, only the register selected for write will be connected. Also, patterns and other decoration of objects may not be included in the imported project.

The following functions and objects will not be converted, or will need reconfiguration:

Function	Comment
Controller	Import of name list and addresses for the first two controllers in the ADP project is supported.
	Import of controller settings such as IP address and COM settings is not supported.
	Import of index registers is not supported.
Alarms	Supported with the following limitations:
Macro and sub-macro	Not supported
Recipes	Not supported

Object	Comment
Action push button	The graphical control will be imported but the actions need to be reconfigured.
Alarm history table, active alarm list, alarm frequency table and alarm marquee	Not supported
Animated graphic	Not supported
Bar graph deviation	The graphical control will be imported but needs to be reconfigured.
Day of week display	The graphical control will be imported but needs to be reconfigured.
Dynamic ellipse and dynamic rectangle	Not supported
Historical data table and historical event table	Not supported
Moving sign	Not supported
Multistate indicator	Not supported
Pie graph	Not supported
Set value push button	Not supported
Seven segments	Not supported
Trend graph	The graphical control will be imported but the curves need to be reconfigured.
X-Y chart	Not supported

Shape	Comment
Arc	Not supported
Free form	Not supported
Parallelogram and solid parallelogram	Not supported
Pie and solid pie	Not supported
Scale	Not supported
Shape	Not supported
Solid polygon	The graphical control will be imported but needs to be reconfigured.

2.4   Optimizing Performance

The following section presents a couple of issues to consider in order to optimize the iX Developer project. Some parts are related to the communication driver; other parts concern CPU load and flash operations.

2.4.1   Communication Performance

Signal Types

Tags used for driver communication can be static or dynamic. These are updated in different manners.

Static Tags

Static tags are updated continuously, even if they are not currently shown on the panel display.

Additionally, the operator panel reads the following items continuously:

- Alarm tags
- Trend viewer tags
- Data logger tags
- Multiple languages tags
- Controller tags with value change events

The communication time is not affected by the following:

- Alarm messages
- Schedulers
- Tags linked to function keys

Dynamic Tags

Dynamic tags are updated only when they are presented on the display. An analog numeric object serves as an example of when a dynamic tag is used.

2.4.2   Communication Design

This section describes how tags are read and how the reading can be optimized to make the communication between the operator panel and the controller fast and efficient.

Keeping Tags in Consecutive Order

Define controller tags consecutively, for example M0.0-M11.7. If the tags are spread (e.g. I0.4, Q30.0, M45.3 etc.) a complete update will take longer time. The number of tags in each package depends on the used driver and information about this can be found in the driver help file.

Tag Packages

Tags to be transferred are not all transferred at the same time, but are grouped into packages. The number of tags in each package depends on the driver for the controller.

To make communication as fast as possible the number of packages should be minimized. Consecutive tags require a minimum number of packages, but it is perhaps not always possible to program it this way. In such cases there is a "waste" between two tags.

Waste is the maximum distance between two tags that can be kept in the same package. The size of the waste depends on the driver used, and is included in the driver help file, e.g as in the table below:

Driver x	Analog signals	Digital signals
Number of signals/package	29	124
Waste	20	0

Bit-Addressed Words vs. Regular Bit Devices

In most drivers it is better to use bit-addressed words than regular bit devices, since you can fit more digital devices in one telegram when using bit-addressed words, than when using bit devices.

Example

Driver x	Analog signals	Digital signals
Number of signals/package	29	124
Waste	20	0

This specific driver can have 29 analog devices or 124 digital devices in one telegram. If you use bit-addressed words in the selected driver, you can fit 464 (29*16) digital devices in one package. This is almost four times as many devices.

ASCII Strings

ASCII strings are transmitted in separate telegrams, and having a large number of strings will affect the communication performance negatively. If an ASCII string only has a small number of different string values, it may be a good idea to use the Text Library function in iX Developer with predetermined contents, thus minimizing impact on driver performance.

Related information
Text Library
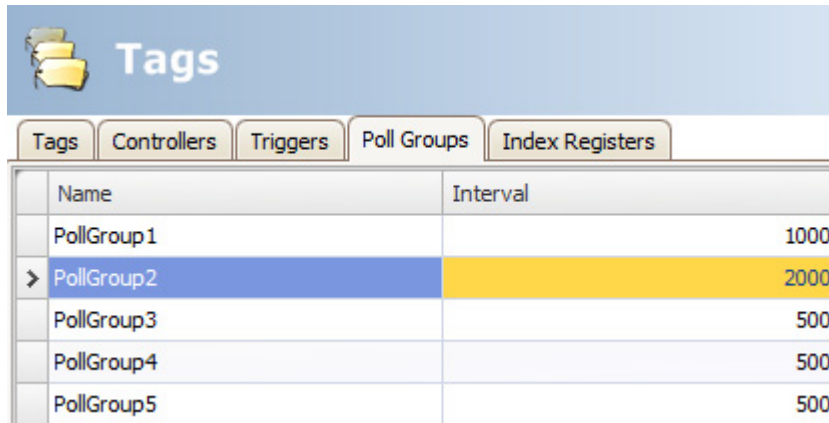
2.4.3 Performance in the Operator Panel

It is important to understand that due to the complex nature of a running system, there are several things that affects the overall performance. The following list gives some examples of things to consider, when it is necessary improve the performance of the operator panel:

- Number of tags
- Number of sampled tags in trend viewers and data logger
- Number of alarms
- Driver performance
- Multiple drivers
- Type of panel
- Alarm list size
- Size of pictures
- Scripts

Poll Groups

The default setting in iX Developer updates all tags every 500 ms. If some tags may be updated less often, it is possible to assign them to a poll group with a lower update interval. Poll groups are configured in the Tags configuration page.

Screens

Optimizing screen update time minimizes CPU load and flash operations. This can make a significant difference in graphic-intense projects that approach the limits of the panel's capabilities.

Screen changes are carried out most efficiently through a show screen action, either on a function key or an object.

Trend Viewers and Data Loggers

Several trend viewers can show data from the same log item.

In order to improve performance in the panel, it is strongly recommended to place all log items logged on the same frequency/event in the same data logger or at least reduce the amount of data loggers.

Try to avoid having one data logger for each trend viewer.

Gradients

Using objects with gradients creates a slightly higher load on the CPU. If a large number of objects use gradients, this will make the screen update time significantly slower. Pay special attention to objects with gradients that are updated frequently, e.g. a bar graph with a style with gradients connected to a tag that changes at short intervals.

Reports

Using the reports function in a project will result in an increased panel startup time.

Related information
Action Groups
Trend Viewer
Reports
Data Logging Strategies
Symbols

2.5 Maneuvering Objects with the Touch Screen

Operator panels with touch screens do not have a built-in keyboard. All maneuvering is done with the touch screen by pressing distinctly with a finger. Because of the resistive touch technology, the screen can only apprehend pressing on one place on the panel at the time. Pressing two points at the same time is interpreted by the panel as a point midway between the two pressed points.

Related information
Keyboard Operated Panels

2.5.1 Operate Objects

For an object to react, it is necessary to specify the behavior, typically by assigning a tag and defining an action. All objects can have actions linked to mouse down, mouse up, mouse enter and mouse leave events. Some objects have a predefined behavior, like sliders setting values, but can also have actions linked to mouse down and mouse up events.

Button

Pressing a button triggers a click- and mouse button event.

Text Box

Pressing on a text box object displays a virtual alpha-numeric keyboard on the panel screen. Type a text using this keyboard and finish by pressing [**ENTER**].

Analog Numeric

Pressing on an analog numeric object displays a virtual numeric keyboard on the panel screen. Enter a value using this keyboard and finish by pressing [**ENTER**].

In runtime a validation of the value is done. If the value entered is out of bounds, the corresponding limit value for the object will be shown.

Slider

Pressing on the desired value position of a slider object sets it to the pressed position.

Related information
Actions Ribbon Tab
Mouse Button Action Triggers
Click Action Trigger
Virtual Keyboard
Validation Group

2.6　Peripherals

2.6.1　USB

External devices such as a USB hub, flash drive, mouse, or keyboard can be connected to the USB host port.

2.6.2　Ethernet

The operator panel has built-in Ethernet port(s) for connection to controllers via TCP/IP. The number of ports varies by model.

Related information
IP Settings

2.6.3　Memory Card

An internal SD memory card can be used to increase project memory in some Operator panels.

An external memory card can be used as storage of e.g. pdf files and media files, to reduce used internal memory. Access to the files on the external memory card is available via scripting.

Related information
Project Group

3   Development Environment

This section describes how to work with iX Developer.

Clicking the **Options** button in the File menu allows adapting the tool behavior.

Related information
File Menu
Options

3.1   Starting iX Developer

Click on **Start/All Programs/iX Developer/iX Developer**.

iX Developer can also be started from the command line.

When iX Developer is started, it is possible to:

- Create a new project using a wizard
- Open an existing project - a list of recently opened projects is displayed
- Open a sample project
- Upload a project from a target
- Import a project created in Information Designer
- View the User's Guide, based on an example project that serves as an instructive introduction to iX Developer
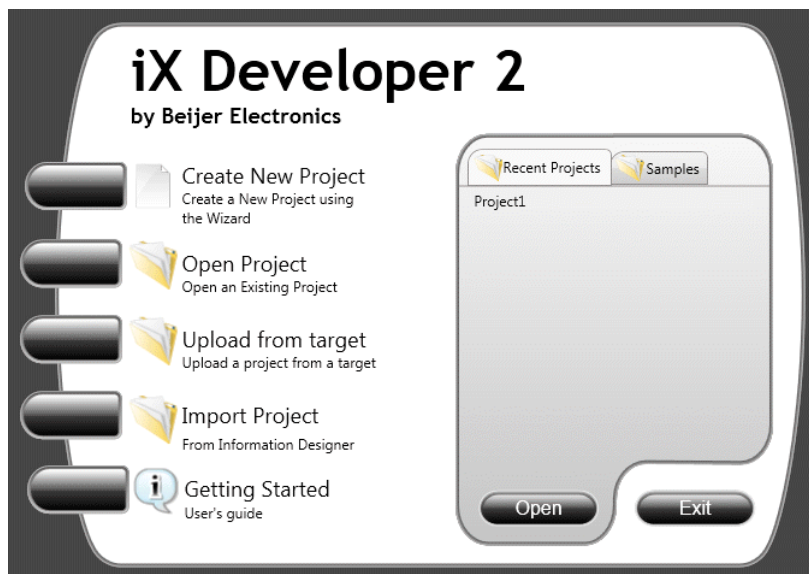
Projects can also be created or opened later from the File menu.

Related information
Creating a New Project
Importing an Information Designer Project
Starting iX Developer from the Command Line
Upload from Target

3.1.1   Creating a New Project

To create a new project using the wizard includes the following steps:

1. Start iX Developer.
2. Select to create a new project.
3. Select target for the project.
4. Select controller brand and model.
5. Name the project and define where files are to be saved in the PC environment.

Select Operator Panel

Select the appropriate target from the presented set of operator panels/PC.

The target can be changed later, from the **Settings** group on the **Project** ribbon tab.

When working with a project for panel T7A, T10A, T7B, T12B or T15B, it is possible to select the rotation of the panel from a drop-down list.

When working with a project for a standard PC, it is possible to select resolution from a drop-down list.

Related information
Changing Project Target

Select Controller

Select the appropriate controller for the project.

The demo controller, including regular tags (data containers) and counters, is used to design and test a project directly on the development PC without connection to an external controller.

Internal tags that work like controller tags, but are independent of an external controller, can be defined.

It is also possible to connect to an external OPC server.

The controller can be changed later by clicking the **Controller** button on the **Controllers** tab of the tags configuration page.

A project can connect to more than one controller. To set up multiple controllers:

Related information
Controller
DEMO Controller
Internal Tags
External OPC Server
Multiple Controllers

Select Location

The **Select Location** dialog controls naming of the project and where project files are saved in the computer environment.
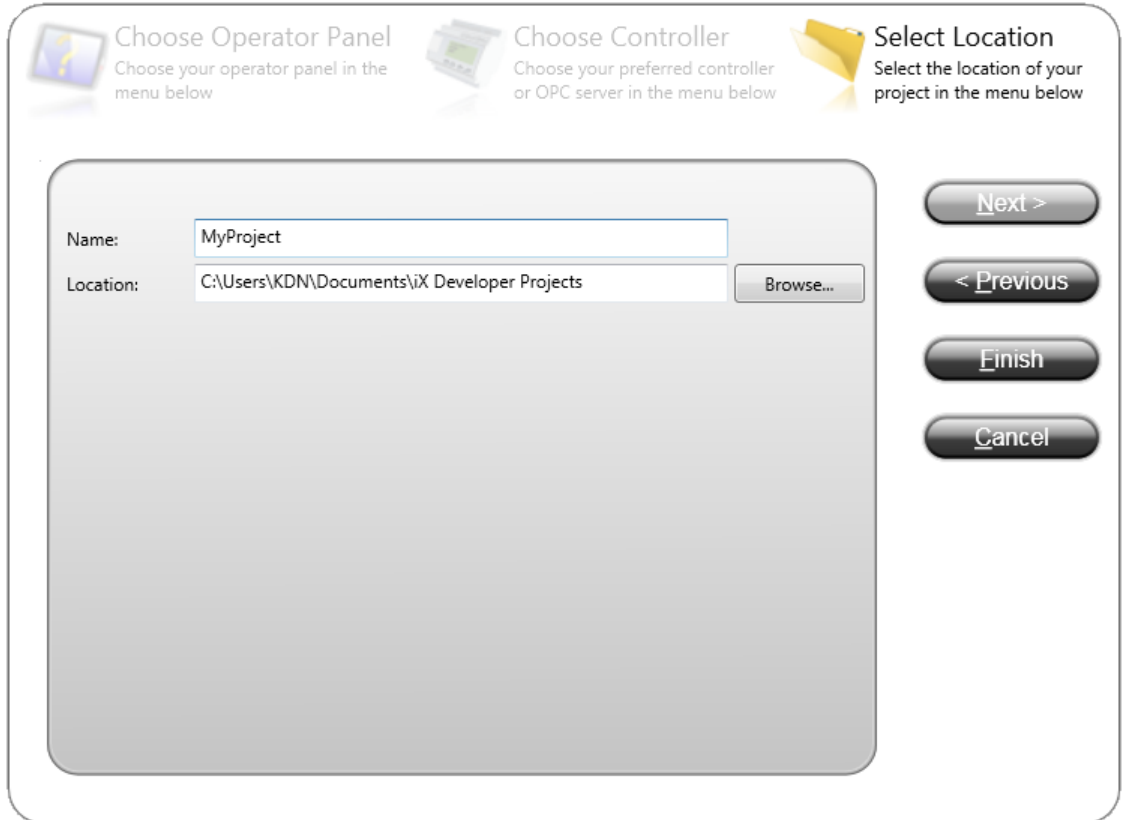
1. Change the project name, if desired.
2. Select where to store the project files by clicking **Browse**, or accept the suggested location.
3. Click **Finish** to create the new project.

3.1.2   Opening a Project

To open an existing project using the wizard includes the following steps:

1. Start iX Developer.

2. Select to open a project.

3. Browse to the project file in the appearing "open project" dialog.

Note:

It is possible to open a compressed ZIP file of the iX Developer project. The user will be prompted for a path where the ZIP file can be decompressed.

Related information
Compress Project
Download

3.1.3　Getting Familiar with iX Developer

The ribbon tabs (e.g. **Home**, **System** and **Insert**) are located in the top of the window when iX Developer is started. The control groups (e.g. **Clipboard**, **Screen**, **Objects** on the **Home** ribbon tab) are available in the ribbon area.
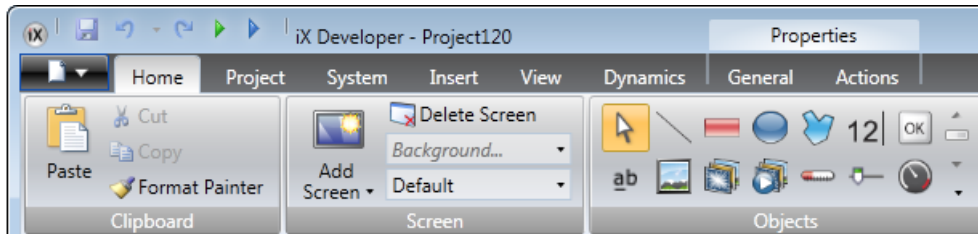
The Project Explorer contains a folder area, and is managed as a docked window.

The desktop area is where screens are drawn and where tags and functions are handled.
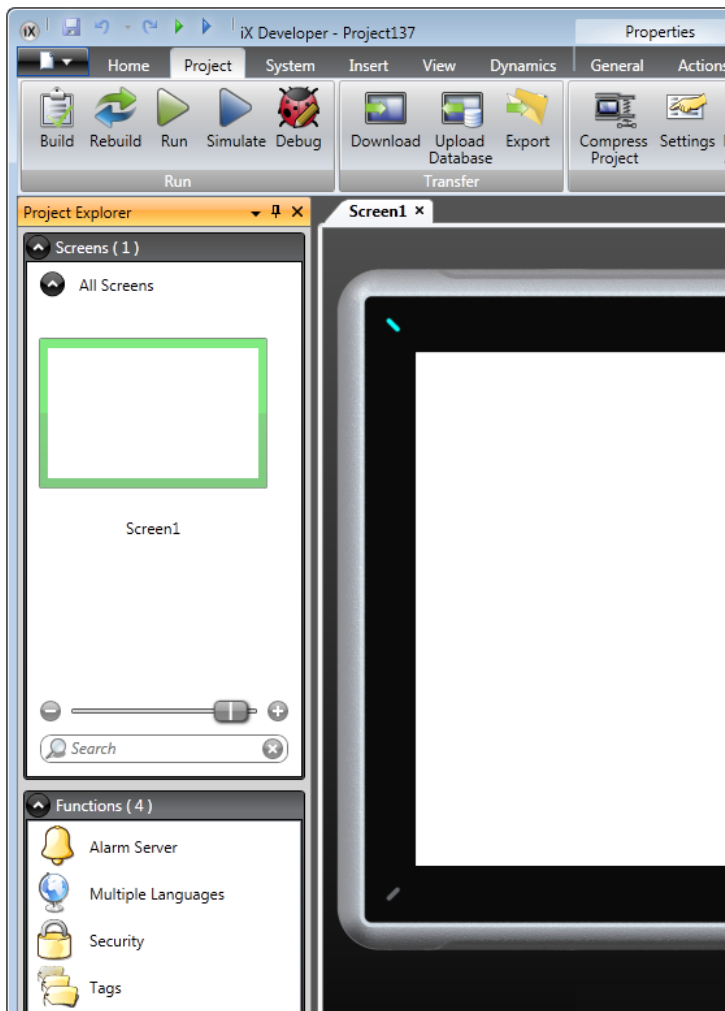
Configuration of objects on a screen is mainly performed using the controls in the **Home** and **General** tabs. Selecting an object on the screen displays the Home ribbon tab; double-clicking displays the General tab, except for the Button object, for which the Actions tab is displayed instead. Several common commands are also available by right-clicking on an object in a screen.

Related information
Project Explorer
Positioning Windows
Screens
Tags

Keyboard Navigation

Pressing **Alt** on the keyboard displays which keyboard shortcut commands can be used to execute a command or to enter a control group, e.g. **Alt + F** opens the **File** menu, and **Alt + I** displays the **Insert** ribbon tab control groups.
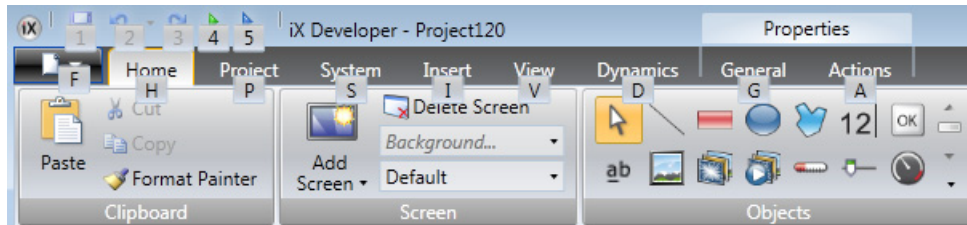
Keyboard Shortcuts and Function Key Commands

The list below includes some of the keyboard shortcuts and function key commands that are available in iX Developer:

Keyboard shortcut / function key	Description
Ctrl + P, [number of the pane]	Switch between panes according to their order on the desktop
Ctrl + E, L	Lock selected elements on active screen (E indicates active editor)
Ctrl + E, H	Hide selected elements on active screen (E indicates active editor)
Ctrl + Z	Undo
Ctrl + Y	Redo
Ctrl + A	Select all
Ctrl + X	Cut
Ctrl + C	Copy
Ctrl + V	Paste
Ctrl + N	New project
Ctrl + O	Open project
Ctrl + S	Save project
Ctrl + B	Bold style
Ctrl + I	Italic style
Ctrl + U	Underline
F1	Display help file. Pressing F1 from within the Script Editor displays scripting help.
F5	Validate project
F6	Run

3.1.4 Starting iX Developer from the Command Line

iX Developer can also be started from the command line, by entering *"[Path to the NeoIDE.exe file]" "[Project location]\[Project name].neoproj"*.

Example

To start iX Developer, and open the project named Test1 in the directory C:\MyProjects, select **Run** from Windows start menu, and type:

- *"C:\Program Files (x86)\Beijer Electronics AB\iX Developer 2\Bin\NeoIDE.exe" "C:\MyProjects\Test1\Test1.neoproj"*

3.1.5   File Menu

The **File** menu contains commands for creating, opening, saving, compressing and closing projects. It also offers the possibility to upload a project from a target and to update the iX Developer software and drivers via Internet or from file. The **File** menu is accessed by clicking on the iX Developer button.
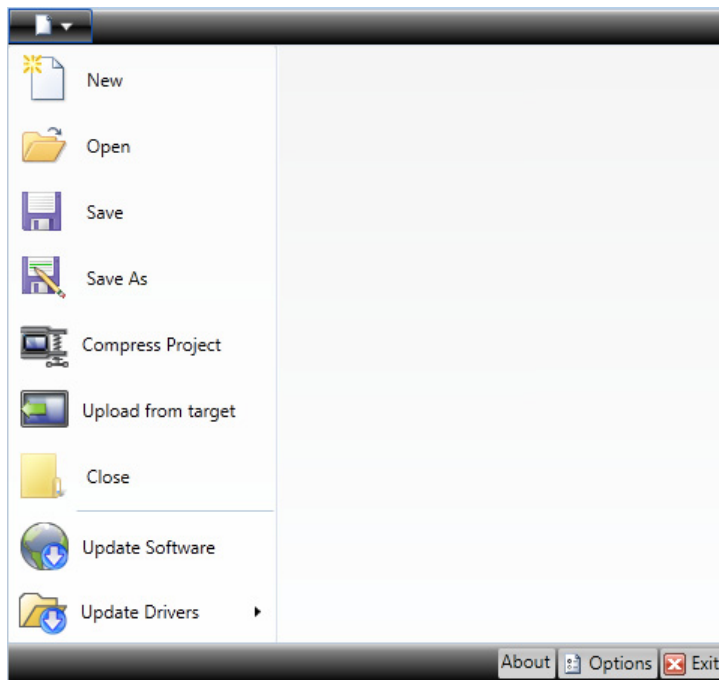
New

When you start iX Developer and select to create a new project, a wizard guides you through the creation of a new project, in the same way as having selected Create New Project from the Welcome page.

Related information
Creating a New Project

Open

Selecting **Open** displays a dialog that allows selection of an existing project file by browsing.

Save

Selecting **Save** saves the project to the current project folder.

Save As

Selecting **Save As** allows saving the project with a different name and/or in a different location.

Creating a new folder when using the **Save as** command helps keeping the large number of project files neatly collected and easy to overview.

Note:
Selecting **Save as** cannot be used in order to make modifications of an existing project, keeping the original project unchanged, and saving the new version with another name. Changes will affect the original project anyway. It is recommended to make a backup copy of the original project, for example by using Windows Explorer, prior to modifying it.

Compress Project

Selecting **Compress Project** compresses the project and saves it as a ZIP file. The user will be prompted for a path where to save the ZIP file and will also be offered the possibility to password protect the ZIP file.

Related information
Compress Project

Upload from Target

Selecting **Upload from Target** allows uploading a compressed ZIP file of a project from a connected target.

Note:
To be able to use the "Upload from target" function, the checkbox "Download all source code" on the "Download" dialog must have been clicked before downloading the project to an operator panel. See the "Download" chapter for more information.

Parameter	Description
Connected targets	Choose the target type to transfer the project from.
Upload	Starts the transfer of the ZIP file from the selected target. The user will be prompted for a path where to decompress the ZIP file. If the ZIP file is password protected, the user must enter the password to be able to upload and decompress the ZIP file. Upload is supported for Panel TxA, TxB, TxC and PC targets. If greyed out, please update to latest "System program".
Verify	The verify function enables the user to check if a target platform configuration is identical to the project configuration. Verify is supported for Panel TxA, TxB, TxC and PC targets. If greyed out, please update to latest "System program".
Log	Show the data being logged during the upload and verify process.

Related information
Download

Close Project

Selecting **Close Project** closes the current project and activates the start wizard.

Update Software

The latest updates and patches for iX Developer can be downloaded directly via an Internet connection.

Update Drivers

Drivers can be updated via Internet or from file.

Related information
Updating Drivers

About

Clicking the **About** button displays information about the current version of iX Developer including the product registration key.

Options

Clicking the **Options** button allows making settings that control the behavior and appearance of iX Developer.

Language Options

Select in which language to run iX Developer . A restart of the application is required for the new language selection to have an effect.

Ribbon Options

- **GoToDefaultRibbonStrategy**: This option changes the active ribbon based on the context of the selection in the desktop. When an object is created or selected, the **General** tab tabs becomes active in the ribbon area. This is valid for any object except for button. A button object will activate the **Actions** tab, as this is considered to be more common as a next choice.
- **SimpleRibbonStrategy**: This is the default setting, and does not swap the active ribbon when the context of the current desktop selection changes. Clicking on an object opens the **Home** ribbon, and double-clicking opens the **General** ribbon for most objects, but for buttons the **Actions** ribbon is activated.

Script Debugger Options

- **Arguments**: The "%1" argument is where the file name will be inserted. This option is applicable only when **Custom** is selected for **DebuggerType**.
- **Debugger**: Full path to the current debugger application.
- **DebuggerType**: Select **Default** or **Custom**. With the **Default** setting, the system will automatically provide the path to the debugger, if found. **Custom** allows the user to provide the path to a specific debugger.

Tags Editor Options

Selections under **Tags Editor Options** makes it possible to customize which columns to show by default in the tags configuration window.

Screen Editor Options

- **EnableInPlaceEditing**: This option allows editing an object directly when selecting it on the screen without having to access its properties on e.g. the **General** ribbon tab.
- **ShowPanelAround**: This option displays an outline of the selected panel model around the desktop area.
- **ShowSizeAndDistanceInformation**: The size of an object is displayed while resizing it. Distance to the edges of the desktop area is displayed when moving an object. The indication can be turned off, if desired, by setting this option to **False**.
- **ShowToolTips**: Useful tool tips are automatically displayed in iX Developer. These can be turned off, if desired, by setting this option to **False**.

The screen editor options can also be modified using buttons in the lower part of the desktop area.

Related information
Screen View in Desktop Area

Mini Toolbar Options

The mini toolbar is a context menu addition with a subset of common controls from the **Format**, **Tag/Security** and **Name** groups.

- **ShowMinitoolbarAutomatic**: When this option is set to **True** the mini toolbar part of the context menu is displayed already when an object is selected. The mini toolbar will always show in combination with the context menu. The mini toolbar will remain active for a selected object, also after it has disappeared, until the selection or the pointer moves away. To show the mini toolbar again move the mouse pointer to the upper right of the object. With the default setting, **False**, the mini toolbar will only show in combination with the context menu for an object.

Exit

Clicking the **Exit** button closes the application. If there are unsaved changes, you are asked if you want to save them before exiting.

3.1.6   Quick Access Toolbar

The **Quick Access Toolbar** contains controls, expected to be used frequently. The toolbar can be customized to include other buttons.

Save

Saves the current project to the location specified when the project was created.

Undo

Executed commands can be undone using the **Undo** button. Clicking the small arrow at the right of the **Undo** button lists recently made changes for selection of the undo operation.
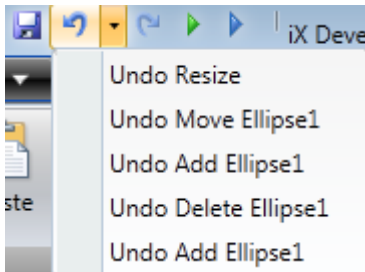
The keyboard shortcut for **Undo** is **Ctrl + Z**.

Redo

A command that has been undone with **Undo** can be redone with **Redo**. The keyboard shortcut for **Redo** is **Ctrl + Y**.

Run

Performs a validation of script code and compiles the project. If the project is valid, a simulation is started on the development PC with connection to the selected controller. Also available from the **Transfer/Test** group on the **Project** ribbon tab.

Simulate

Performs a validation of script code and compiles the project. If the project is valid, a simulation is started on the development PC without connection to the selected controller. Also available from the **Transfer/Test** group on the **Project** ribbon tab.

Related information
Project Ribbon Tab

Customizing Quick Access Toolbar

Some of the ribbon controls can be added to the Quick Access Toolbar. Right-click on the desired control and select **Add to Quick Access Toolbar** or **Remove from Quick Access Toolbar**.

3.1.7 Ribbon Tabs

The ribbon tabs contain grouped controls for editing and maintaining a project, screens and objects. Click on any ribbon tab to display the ribbon.

Use **Minimize the Ribbon** and **Maximize the Ribbon** to collapse and expand the ribbon control tabs, or double-click on a ribbon tab. This can also be done from the Quick Access toolbar.

Related information
Home Ribbon Tab
Project Ribbon Tab
System Ribbon Tab
Insert Ribbon Tab
View Ribbon Tab
Dynamics Ribbon Tab
General Ribbon Tab
Actions Ribbon Tab
Quick Access Toolbar

3.1.8   Additional Properties

Some ribbon groups include additional properties that can be configured by clicking the small arrow in the lower right corner of the current group.
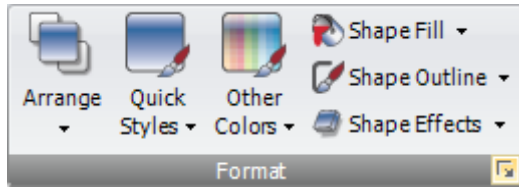
The properties window displays available property groups available for selection at the left. The property groups available differ depending on which object is selected.
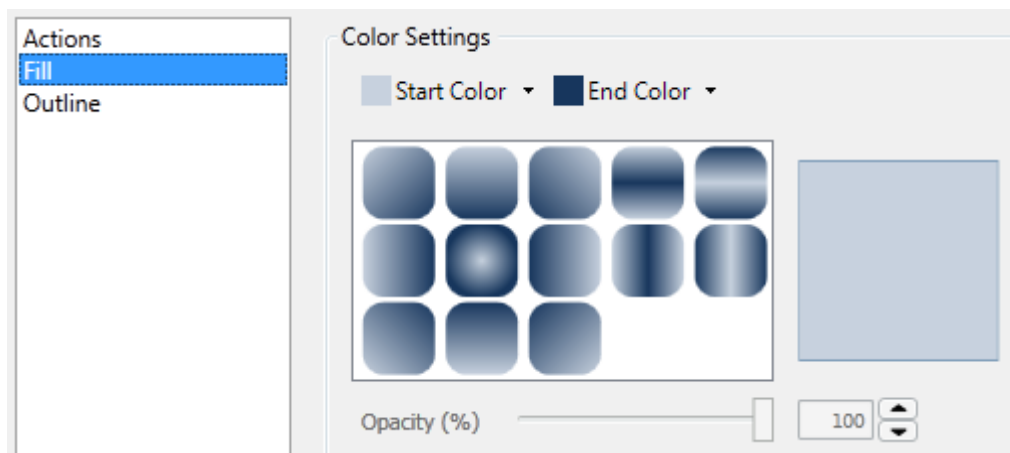
3.2 Desktop Area

The desktop area displays screens and configuration pages for project components such as controllers and functions. The desktop area shows only one screen or component at a time. When multiple screens or components are opened a row of tabs are shown in the upper part of the desktop area. Clicking on a tab activates its contents for editing.

If there are more tabs open than can be displayed, navigation arrows in the upper part of the desktop area can be used to scroll between them.

3.2.1 Screen View in Desktop Area

Object appearance as well as size and appearance of the current screen can be managed by controls in the lower part of the desktop area. It is also possible to turn on and off the in-place editing function that allows editing text in an object directly when selecting it on the screen.

Show/Hide Size and Distance Information

The size of an object is displayed while resizing it. Distance to the edges of the desktop area is displayed when moving an object.
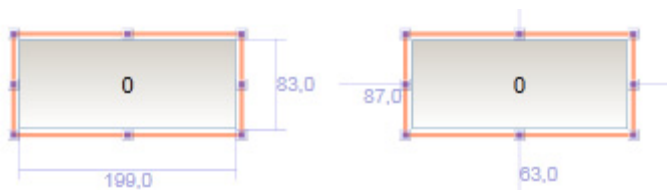
Show Size and Distance Information is activated

Show/Hide Info

It is possible to show information about which tag each object is connected to, and if dynamics or actions are configured for the object, by clicking on the **Show/Hide Info** button.
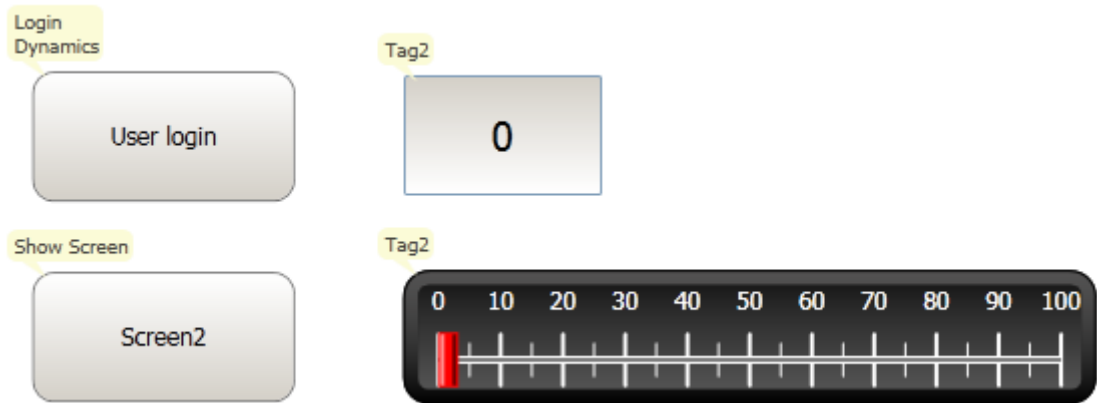
Show Info is activated

Show/Hide Screen Tooltips

A semi-transparent tooltip is displayed for each object when the cursor is rested on it, if **Show Screen Tooltips** is selected. Tooltips for e.g. menu commands are not affected by this setting; only tooltips for objects on the screen.
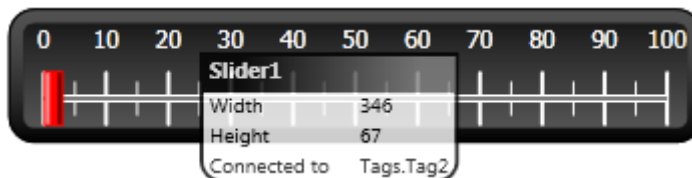
Show Screen Tooltips is activated

Enable/Disable In-place Editing

This option allows editing an object directly when selecting it on the screen without having to access its properties on e.g. the **General** ribbon tab.
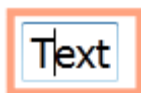
In-place Editing is enabled

Show/Hide Panel Frame

Show/Hide Panel Frame displays or hides a virtual panel frame surrounding the screen, when an operator panel is selected as target.

Fit to Screen

Click on **Fit to Screen** to make the screen the maximum size that will fit in the desktop area.

Zoom

The following methods can be used to zoom in to or out from the desktop area:
- Selecting a set zoom value from the **Zoom** drop-down list.
- Setting the **Zoom Slider** to a desired zoom level.
- Clicking the **Zoom In** and **Zoom Out** buttons.
- Placing the mouse pointer anywhere in the screen and using the scroll wheel.

Pan

Panning of the current screen is possible in Layout mode, if the screen is larger than the desktop area.
- Press and hold [**Spacebar**] (the mouse pointer will change its appearance from an arrow shape to an open hand).
- Drag to pan the screen (the mouse pointer hand closes in panning mode).

3.2.2   Desktop View Modes

Screens and functions have different view modes. The default setting is **Layout** view mode, but screens and functions can be partly viewed and edited directly in code. The desired mode is selected from the lower left part of the desktop area. The code for screens is in xaml and in C# for script.

| Layout | Xaml | Script | Aliases |
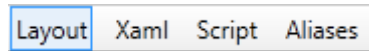
Desktop view mode tabs

Note:
Incorrect code entered in text mode (Xaml and Script) may result in errors that are not possible to recognize during analysis and build.
Erroneous code may result in unpredictable behavior and loss of data.

Layout

Screens and objects are usually edited in **Layout** view mode.

Xaml

A screen layout is defined in Xaml (Extensible Application Markup Language) code. The .xaml files are stored in the project folder.

Xaml code can be added and changed in the **Xaml** view mode. Any changes done in the Xaml code will affect the screen layout.

Script

In **Script** view mode scripted actions can be configured for screens, objects, tags, function keys etc.

Note:
To display a script guide that describes iX Developer scripting possibilities in detail, select **Script** view mode, click anywhere in the code, and press **F1**.

Objects that can trigger events will show with expandable nodes in the **Script** view. Clicking the [+] of such a node expands it and shows the trigger events for the object. Double-clicking on an event inserts its method heading with an empty code body. Scripts are written in C# syntax.

When actions are defined for an object, this will have precedence over script code.

Note:
Avoid using multiple trigger methods as this may lead to undesired behavior.
Use either a script or define actions with the controls on the **Actions** tab. Avoid using *Click Action Trigger* actions in combination with *Mouse Button Action Triggers*.

Note:

The **ValueOff** and **ValueOn** events for internal variables are available for selection in script mode, but they are not executed when running the project.

Note:

The **ValueChangeOrError** event is available in script mode, but it should only be used with controller tags. It does not support internal variables or system tags.

Complex behavior can be created using script code.

Selections made in **Layout** view mode are kept when switching to **Script** view mode and vice versa.

Nodes with edited script code are highlighted in the expanded tree view.
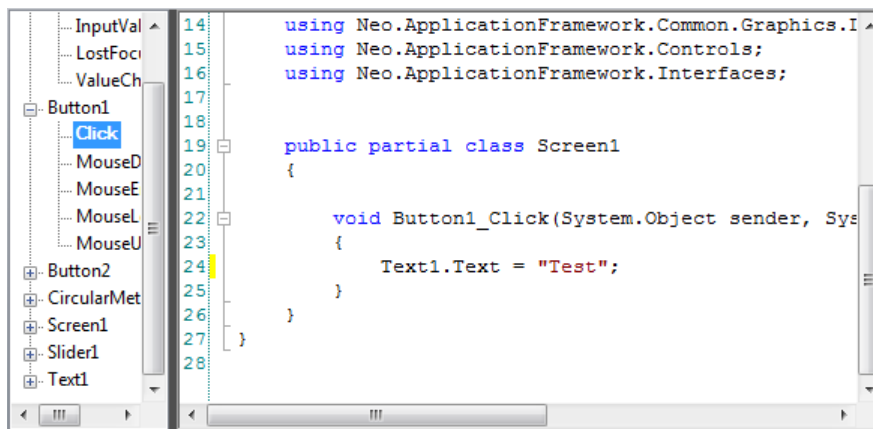
Highlight of an edited node in the tree view

It is possible to disable a script without deleting it, by right-clicking on a node with script, and selecting **Unhook event handler**. This excludes the script from being run, even though there is code, and the node will no longer be highlighted.
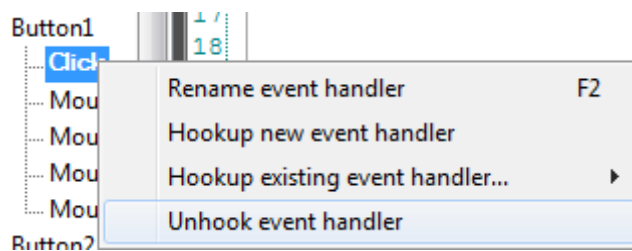
Some objects, such as text boxes, list boxes and combo boxes, rely on script code for advanced behavior.

Name Completion

A context sensitive name completion feature (IntelliSense) can be activated during typing with [**Ctrl**] + [**Spacebar**] and it triggers automatically when a period ('.') is typed after a code element. Using items outside the current scope is defined with the keyword **Globals**, but it is not possible to script to other screens than the currently selected.

Methods are completed without argument lists. Add parenthesis and arguments to complete a selection.

Note:
Incorrect script code may result in errors that are not possible to recognize during analysis and build. Erroneous code may result in unpredictable behavior and loss of data.

If there are errors in script code during validation of a project, it is possible to navigate to the erroneous position by double-clicking the error in the error list.

Press [**Ctrl**] + [**Shift**] + [**Spacebar**] while editing to get a tool tip with the allowed variants of an overloaded method. Use [**Up arrow**] or [**Down arrow**] to scroll. Make a new selection or press [**Esc**] to close the tool tip.

Note:
It is necessary to do an explicit type casting of the tag format for operands of an overloaded method.

Script code can be used to format objects. If an object attribute is selected a tool tip with allowed methods will appear. Methods not part of the default C# API is either addressed with the full path or the library can be included.

Related information
Error List
Tag Format
Format Group

Example:

Filling a rectangle object using a color gradient from one color to another color.

```
void Screen1_Opened(System.Object sender, System.EventArgs e)
{
    Rectangle1.
}
```
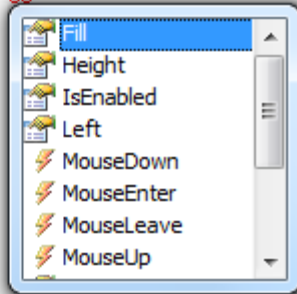
The following code implements a gradient from red to purple in the rectangle.
The library with the "BrushCF()" method is added with "using" in the beginning
of the example, together with the default libraries.

```
public partial class Screen1
{

    void Screen1_Opened(System.Object sender, System.EventArgs e)
    {
        Rectangle1.Fill = new BrushCF(System.Drawing.Color.Red,
            System.Drawing.Color.Purple,
            FillDirection.Center);
    }
}
```

Note:
When a timer is used in script code, it is important to close the timer in the code.

Note:
Renaming objects and screens included in scripts in Layout mode will cause the scripts
not to function.

Referenced Assemblies

Third-part components and objects that are added to the project via **Referenced
Assemblies** will be made available when scripting. Their methods and properties
will be included by the name completion function, in the same way as for built-in
components.

Related information
Referenced Assemblies
Name Completion

Aliases

Aliases can be configured in the **Aliases** view mode.

Related information
Aliases

3.2.3   Positioning Windows

The behavior of many windows can be adapted in order to suit the user; for example shown or hidden, or moved to another position on the desktop.

Docked windows representing control tools can be converted to floating windows. Control tools can be docked separately or be managed as a tabbed group in a window. A docked tool window supports auto-hide.

When a window is dockable, a guide diamond appears when the window is moved on the desktop. Extended position arrows appear at the desktop edges. The guide diamond makes it possible to dock a tool window to one of the four sides of the desktop area. When a tool window is undocked, it floats on top of the desktop. Dropping a tool at the center of a guide diamond creates a tabbed group of the tools inside the window that the guide diamond belongs to.

Docking a Tool

Drag the window heading to any position on the screen. If the tool is not docked it will become floating.

To re-dock a tool:

1. If the tool is docked already, double-click on its heading to change to floating.

2. Drag the tool to the center of the desktop. A guide diamond appears, pointing towards the four edges of the desktop. Release the window to the desired direction. When the mouse pointer is in position, an outline of the window appears in the designated area.

Docking a Tabbed Group

It is possible to dock a tool to an existing group of tools, creating a grouped window with tabs:

1. If the tool is docked already, double-click on its heading to change to floating.

2. Drag the tool to the center of the existing group. A guide diamond appears.

3. Drop the tool at the guide diamond center.
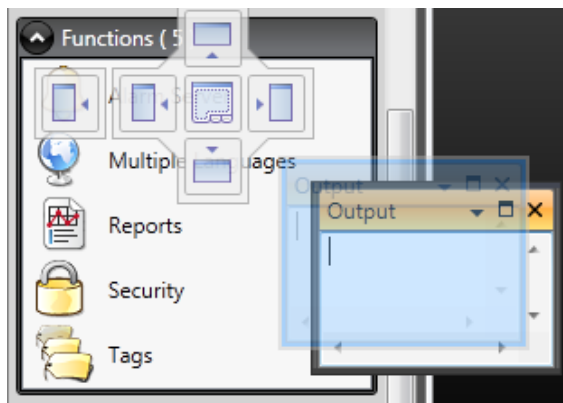
Releasing a Tool from a Tabbed Group

When a window contains multiple tools, they will appear as tabs in the lower part of the window. To release a tool that resides in a tabbed group, drag the tab for the tool and drop it on the desktop.

Auto-Hide

Auto-Hide enables a maximized view of the desktop area by minimizing tool windows along the edges of the desktop area when not in use. Auto-Hide is turned on and off by clicking the Auto-Hide pushpin icon on the window title bar.

Tool windows will expand to original size when the pointer hovers over them. When a hidden tool window looses focus, it slides back to its tab on the edge of the desktop area. While a tool window is hidden, its name and icon are visible on a tab at the edge of the desktop area.

Note:
The Auto-Hide function has to be turned off before a tool can be moved.

3.2.4   Configuration Pages

Many functions, such as Alarm Server and Data Logger, are configured via table style configuration pages.

In some cases there are multiple configuration pages for a feature. When applicable, the configuration page for a feature can be opened to allow direct access. For example, it is possible not only to select a tag, but also to declare new tags, directly through the **Select Tag** field for an object.

Home

In a configuration page, the **Home** tab is used to add and remove items in a function or controller.

Edit Table Cells

Use the **Add** button to append a new item in the table view of the configuration page.

Use the **Delete** button in the **Home** tab or the **Delete** button on the keyboard to remove the active row in the table. Multiple rows can be selected using the [**Ctrl**] or [**Shift**] keys.

Keyboard Navigation

Navigation in configuration pages can be made using arrow keys or with [**Tab**]. Pressing **F4** inside a table cell will in some cases open selection list.

Pressing [**Enter**] acknowledges a selection, and pressing [**Esc**] cancels a selection.

Press [**Delete**] to remove a value when editing in a field. If a row is selected when pressing [**Delete**], the entire row will be removed from the table.

Use [**Spacebar**] to toggle a value in a check box.

Press [**Ctrl**] + [**Tab**] to shift focus from the table.

Navigation in Layered Pages

A layered configuration page (for example the alarm items page) has some specific navigation functions.

Keyboard navigation down is made by pressing [**Ctrl**] + [**Down Arrow**]. This will navigate into an expanded group, and out again when at the last of the expanded rows.

Keyboard navigation up and out of a group is done by pressing [**Up Arrow**].

Expand a group with [**Ctrl**] + [**+**], collapse with [**Ctrl**] + [**-**].

Keyboard navigation between fields inside a group is made with [**Right Arrow**] or [**Left Arrow**].

Pressing [**Enter**] acknowledges a selection, and pressing [**Esc**] cancels a selection.

Auto Fill

Fields in the configuration tables are when possible automatically filled in when [**Down Arrow**] is pressed when at the last table row. When needed, names and tag identifiers are incremented to avoid exact copies of symbolic names and tag identifiers. Incrementing is done if the last part of the string is an integer number. A "1" will be appended to the string if no numeral character ends the string. This is useful when there are consecutive tags of the same type in the controller data.

Filtering Items

Items in configuration pages can be filtered by selecting **Show Selection** from the **Show Selection/Show All** button.
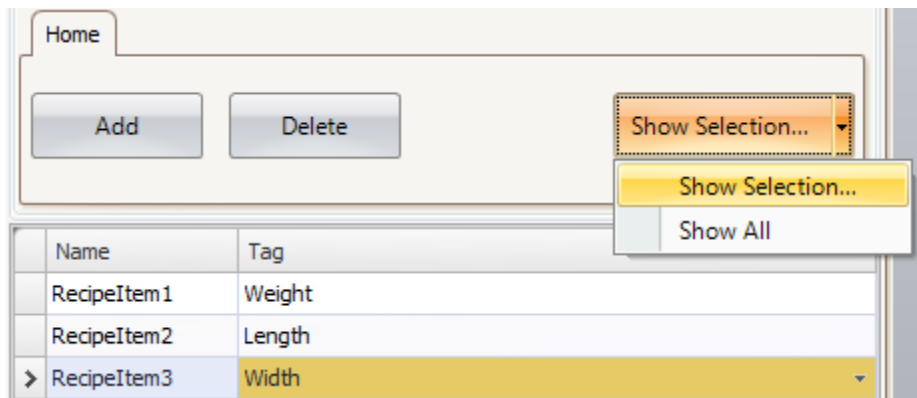
Clicking the button opens the Filter Builder.
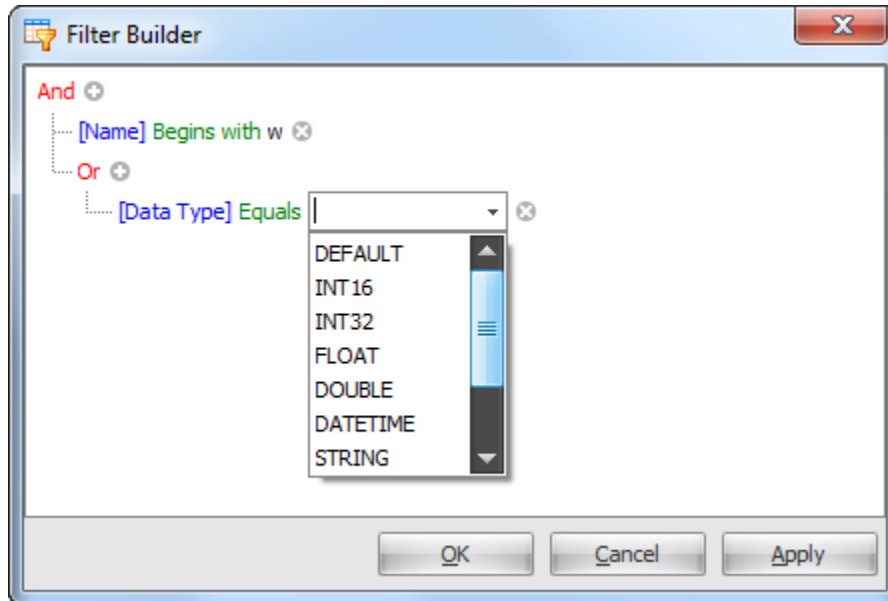
Several conditions can be used, and the different values/operators that can be used for making conditions are displayed when clicking on the condition building blocks.

Operands and operator nodes are context sensitive. Name strings are not case sensitive. Values in the filter builder are saved individually for each function during an edit session. A filter is built as a logical expression tree. A filter typically has operations on individual column values as leaf nodes and logical operators serve as main nodes.

The following color scheme is used:

Color	Function
Red	Logical operator
Blue	Column operand
Green	Value operator
Grey	Value operand

An indication that a filter is used is displayed at the lower part of the configuration page.
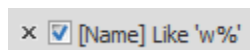
Selecting **Show All** from the **Show Selection/Show All** button in the configuration page displays the complete, unfiltered list.

3.3   Screens

Screens contain objects that can be displayed for the operator in runtime. A screen can contain predefined objects, such as buttons or an alarm list. Pictures of physical objects, such a pump, connected to controller data, can also be included to be used for control and monitoring of the physical object.

Properties are defined for each screen.

Screens are edited in the desktop area and can be managed from:

Location	Used for
The Navigation Manager	Adding new screens, managing links between screens, deleting screens
The Screens list in the Project Explorer	Adding new screens, importing screens from other projects, renaming screens, setting screen as startup screen, saving screens as templates, deleting screens, editing screen scripts
The Screen group of the Home ribbon tab	Adding new screens, selection of screen template and background screen, deleting screens, language selection, screen security
The Screen group of the General ribbon tab	Managing popup screen properties and selecting screens for preloading

3.3.1   Screen Name and Screen Title

All screens have unique names, visible e.g. in the Project Explorer and the Navigation Manager. Names of all open screens are shown as tabs in the desktop area. The name of the current screen is displayed and can be changed in the Name group of the Home tab.

In runtime, the screen name is displayed together with the project title in the title bar as default. To display something other than the screen name, a screen title may be added in the Screen group on the General tab when the screen is selected.
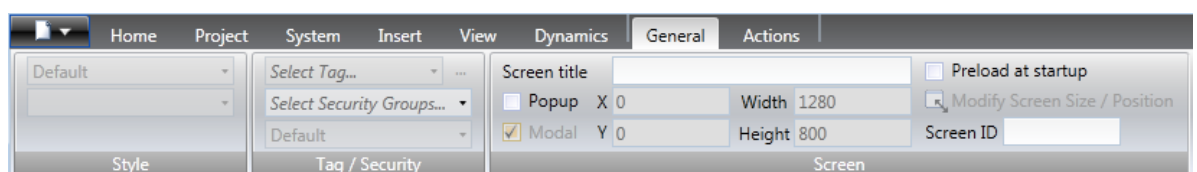
If the project title on the Project tab is left empty, only the screen name or screen title is displayed.
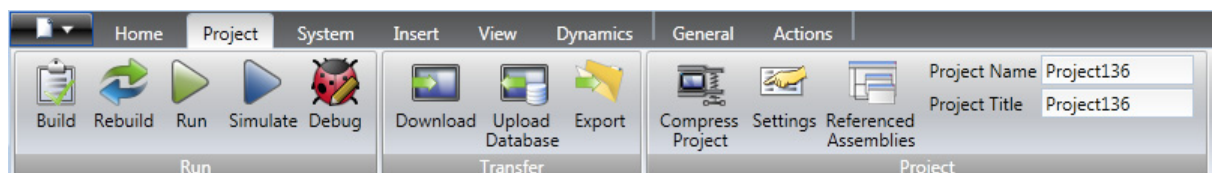
3.3.2   Background Screen

It is possible to use another screen as background screen, to, for example, create a uniform design of the screens in the project and avoid repetitive configuration.

Background screens belong to the current project.

Example

Follow the steps below to use a background screen:

1. Create a background screen with some reusable elements, such as navigation buttons, and rename it to a suitable name.
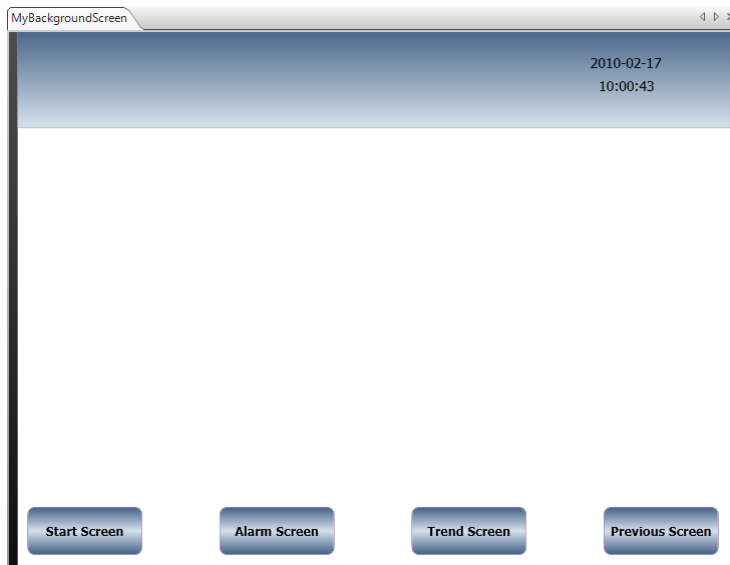
2. Create a new screen.

3. Keep the new screen open on the desktop, and select the background screen from the **Screen** group of the **Home** ribbon tab.
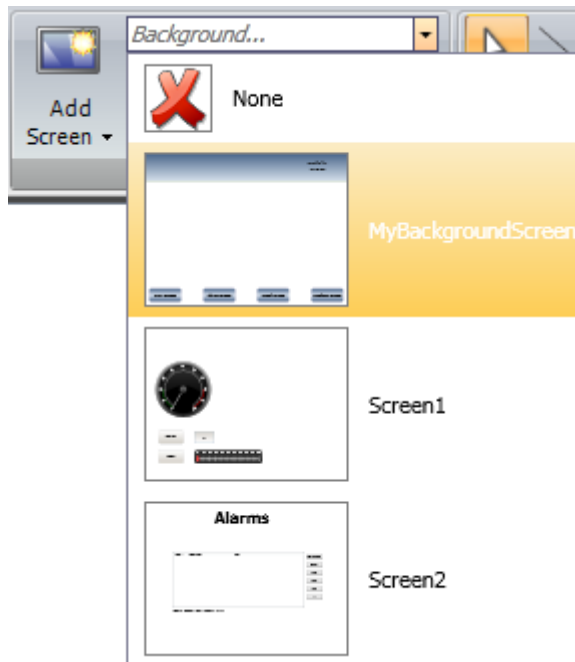
The background screen is now used for the screen.

Any modification made to the background screen will be reflected in all screens using it.

3.3.3 Startup Screen

The startup screen is the first screen that opens when a project is executed. By default, Screen1 is set as startup screen.

Another startup screen can be set at any time by right-clicking on the screen on the desktop or in the Project Explorer and selecting **Set as Startup Screen**.

The selected startup screen is easily distinguished from the other screens in the Project Explorer by a blue outline when in focus.

3.3.4 Screen Template

The installation of iX Developer includes a number of screen templates with basic functionality. It is also possible to save your own screens as screen templates, by right-clicking on the current screen and selecting **Save Screen as Template**.

The screen template can be used in the current or other iX Developer projects.

If a selected screen template contains tags that were not previously included in the project, you can select to automatically add these to the Tags list.

To use a template screen, click the lower part of the Add Screen button in the Screen group of the Home ribbon tab. A preview of screen templates is displayed.

User-created as well as predefined screen templates may be deleted by right-clicking on the template in the preview and selecting **Delete Screen Template**. The folder of the selected screen is then permanently deleted from disk.

3.3.5 Screen Security

Using security settings for screens makes it possible to restrict access to screens based on user security groups, similarly to object security.

It is not possible to set security for the Startup screen.

When using a background screen, the security settings of the top level screen are applied.

When screens are imported from other iX Developer projects or from Information Designer, security settings are included and will be used in the current project.

Related information
Object Security and Visibility

3.3.6 Popup Screen

Normally, only one screen is visible in the operator panel in runtime. Sometimes, a popup screen that behaves like as a floating window may be useful. Any screen can be configured as a popup screen:

1. Check **Popup screen** in the Screen group of the General tab.
2. Set the start position coordinates (X, Y) to determine where the screen will open. The position (1,1) corresponds to the upper left corner of the screen.
3. Set the desired width and height in pixels of the screen resolution.

Clicking **Modify Screen Size/Position** allows changing the popup screen size and position by dragging it with the cursor. The coordinates are updated automatically.

Modal Popup Screen

Checking the **Modal** option makes the selected popup screen modal, which means that the user cannot interact with other screens while this screen is visible. All other screens will still be updated.

Style

Different styles can be selected for popup screens for PC targets.

Note:
When using a web browser or pdf viewer object in a popup screen, only the default style for the popup screen is supported.

Open Popup Screen

A popup screen is opened in runtime by a **Show Screen** action on an object.

Note:
When configuring a Show Screen action for a popup screen, position of the screen can be selected. These settings override the settings made in the Screen group of the General tab.

Close Popup Screen

A popup screen is closed in runtime by a **Close Screen** action on an object.

3.3.7 Preloading Screens

The first time a screen is opened in a running project, some time elapses while compiling it. In order to minimize this time, it is possible to preload a maximum of 10 screens from the **Screen** group on the General tab while the screen is selected. The selected screens will then be loaded at application startup instead of when opening the screen.

Note:
The time saved at screen opening is instead added to the application startup time.

Events related to screen actions (Screen Opened, Screen Closed and Screen Closing) will not be carried out when a screen is preloaded.

Preloading the startup screen (or its background screen) has no effect. Popup screens cannot be preloaded.

Data items used by a screen selected for preload will be enabled during preloading and then disabled.

The preload function is available for operator panel targets only.

3.3.8 Importing Screens

Screens from other projects can be imported, by right-clicking in Screens area of the Project Explorer and selecting **Import**.

If the selected screen has a name identical to one of the existing screens, a new screen name is proposed.

If the imported screen includes tags that do not already exist in projects, you are asked if you want to add these.

Note:
No tag information, other than tag names, will be added by the import function.

3.3.9 Grid

There are different methods to align objects in screens; **Snap objects to other objects** and **Snap objects to grid**. Grid properties are managed by right-clicking on the current screen and selecting **Grid** from the context menu. It is also possible to select not to snap objects.

Grid settings are global and will affect all screens.
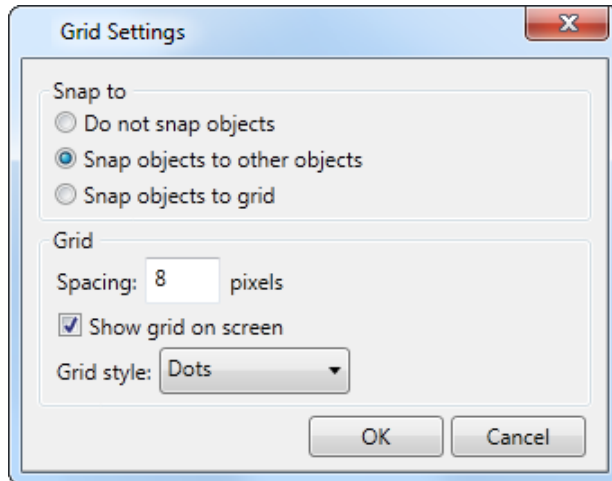
Snap Objects to Other Objects

To help position objects in a screen snap lines will appear when an object is moved across the screen. The snap lines are used to align the dragged object with other objects on the screen.

Blue snap lines appear whenever the right, left, top, or bottom boundaries coincide with other object boundaries. A horizontal red snap line indicates that the center of the object is aligned with the center of another object. For vertical center alignment, select the objects and use **Align Center** in the Arrange control of the Format group.

Snap Objects to Grid

Use the context menu and select **Snap to Grid** to enable binding to grid points for a moved object. **Snap to Grid** mode disables **Snap Lines** for dragged objects.

Grid

The different grid styles can be used as a help pattern when doing the layout of the objects on the screen.

Grid setting	Description
Spacing	Number of pixels between grid lines
Show grid on screen	Displays the grid on the screen
Grid style	Selection between dots and lines as grid style

3.4 Objects

Objects that can be created in a screen are available from the Objects group and the Component Library.

Snap lines are used to position objects in a screen and align a moved object with other objects.

Information about tag connections as well as size and distance information for objects on the current screen can be displayed by using the buttons in the lower part of the desktop area.

Related information
Objects

3.4.1 Handling Objects

Create

To insert a default-sized object, click the position on the screen where you want to place the object.

To give the object a specific size and shape, press the mouse button and drag to create the object on the screen. Press and hold [**Shift**] while creating an object to retain the aspect ratio of the object.

When creating a line object, press and hold [**Shift**] to constrain the line at 45 degrees angles.

Select

The **Select** tool is available at top left in the Objects group when the group is not expanded. The Select tool is the default tool when editing screens. After applying any other object control, the mouse pointer will return to the Select tool. When the Select tool hovers over a selected object the mouse pointer changes to a cross with arrowheads.

Move

The position of an object is part of the properties for the object. The position of an object can be changed in different ways:

- Click and drag an object to move it to the desired position.
- Open the property grid and edit the **Left** and **Top** coordinates.

To move the selected object one pixel at the time, select the object and use the arrow keys on the keyboard.

Copy and Move

Use [**Ctrl**] + drag to make a copy of the selected object, and to move it to the desired position.

Resize

The size of an object is part of the properties for the object. To change the size of an object:

- When the Select tool hovers over the resize handles of a selected object, the mouse pointer changes to an arrow. Resize the object to the desired size.
- Open the property grid and edit the **Height** and **Width** values.

To resize the selected object one pixel at the time, select the object, press [**Shift**] and use the arrow keys on the keyboard.

Rotate

Note:
Rotate is only supported for projects designed for PC targets, and not for all objects.

Angle and rotation center is part of the properties for an object. The rotation center for an object is shown as a small white circle which by default is located in the center of the selected object.

When the Select tool hovers over the rotation center of a selected object the mouse pointer changes to a circle inside a square with arrow sides. This indicates that it is possible to move the rotation center.

To change the rotation center of an object:

1. Point to an object to select it. When an object is selected a red frame with handles is shown around the object.

2. Drag-and-drop the rotation center circle to the position that is to be the center of the rotation.

When the Select tool hovers next to a corner of the selection frame for a selected object the mouse pointer changes to a circular arrow. This indicates that the object is possible to rotate by the frame handle around the rotation center.

To change the rotation angle of an object:

1. Drag to rotate the object around its center position.

2. Use [**Ctrl**] + drag to rotate in steps of 15 degrees.
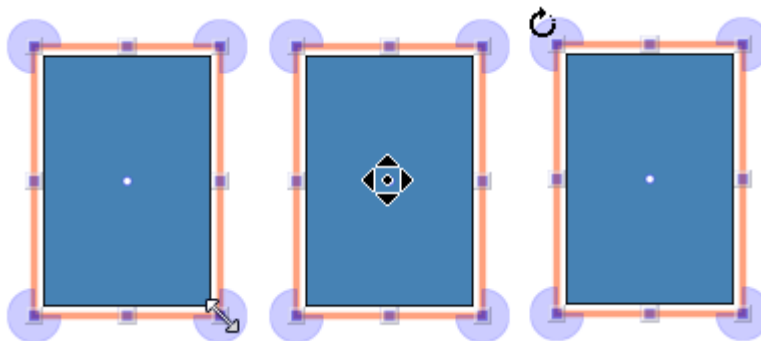
Cursor appearance for Resize, Rotation centre, and Rotate.

Selecting Multiple Objects

Multiple objects on the current screen can be selected at the same time, with the use of the mouse pointer to draw a frame that touches the objects. Alternatively, select one object and then use [**Shift**] + click to select or de-select other objects.

A multiple selection can be used to arrange objects, by using the Arrange control on the Format ribbon tab. The object with the primary selection is used as guide object.

When multiple objects are selected, the primary selection is shown with a red frame, while all other objects in the group have blue frames.

Click on any object in the group to change this to the primary selection. A click or double-click on any object will make it the primary selection, and, if applicable, change the ribbon focus according to the selected ribbon behavior.

The groups shown in the General and the Actions tabs are dependent on the selection. When multiple objects are selected the tabs will show only the properties that are common for the objects in the selection. Property changes will affect all selected objects.

Hiding and Locking Objects

Objects can easily be hidden and locked using the Object Browser, but the commands are also available from the context menu. To make a hidden object visible again, or to unlock a locked object, the Object Browser can be used.

Related information
Object Browser
Property Grid

Creating Series

The Create series function facilitates creating a number of identical objects. The function handles adding graphic elements as well as controller connections.

The following example creates a series of analog numeric objects quickly and easily:

1. Create an analog numeric object and connect it to a tag (in this example; D0).

2. Right-click on the object and select **Create series** from the context menu.

3. Fill in the number of objects to add and make settings for the address incrementation, and click **OK**.
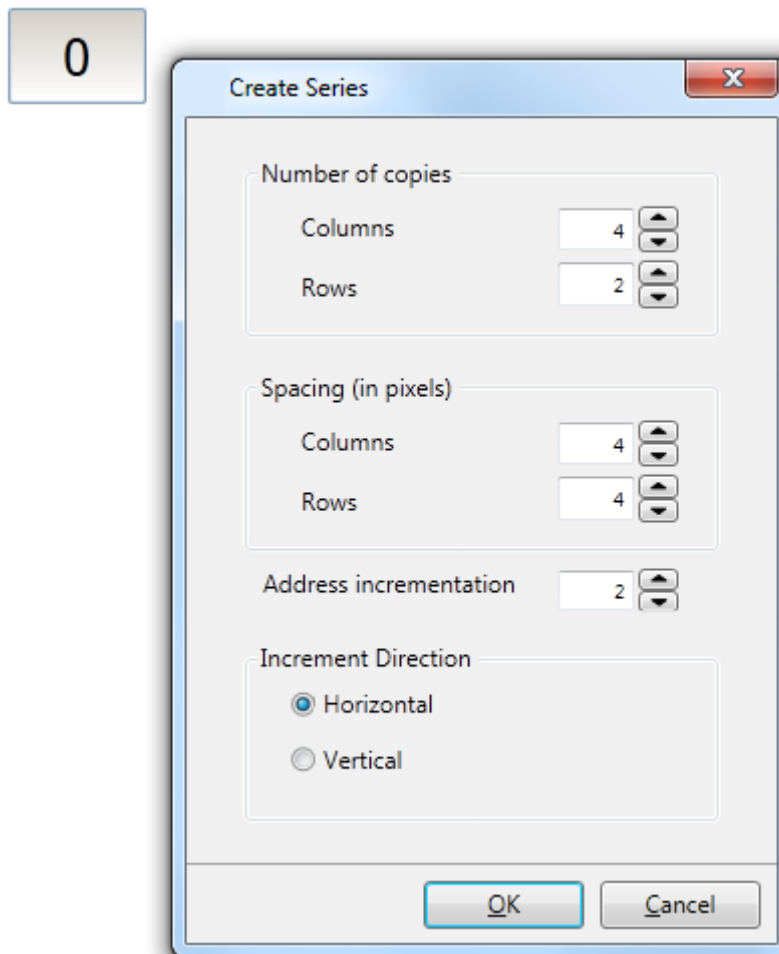
Parameter	Description
Number of copies	Enter the number of columns and rows to create
Spacing	Specify the number of pixels between the added objects
Address incrementation	Select how to increment addresses for the added objects. Only addresses available in the driver will be added.
Increment direction	Select if addresses are to be incremented in horizontal or vertical direction

Graphical objects are added on the screen according to the settings made:

Tags are added to the project according to the settings made:

Right-clicking in the Tags configuration page and selecting **Create series** makes it possible to create a series of tags quickly and easily, without adding graphic objects.

3.5 Navigation Manager

The Navigation Manager is used to manage screens and their relations with respect to project navigation in runtime. It can be activated/deactivated from the View tab. It provides an overview of all screens in the current project.
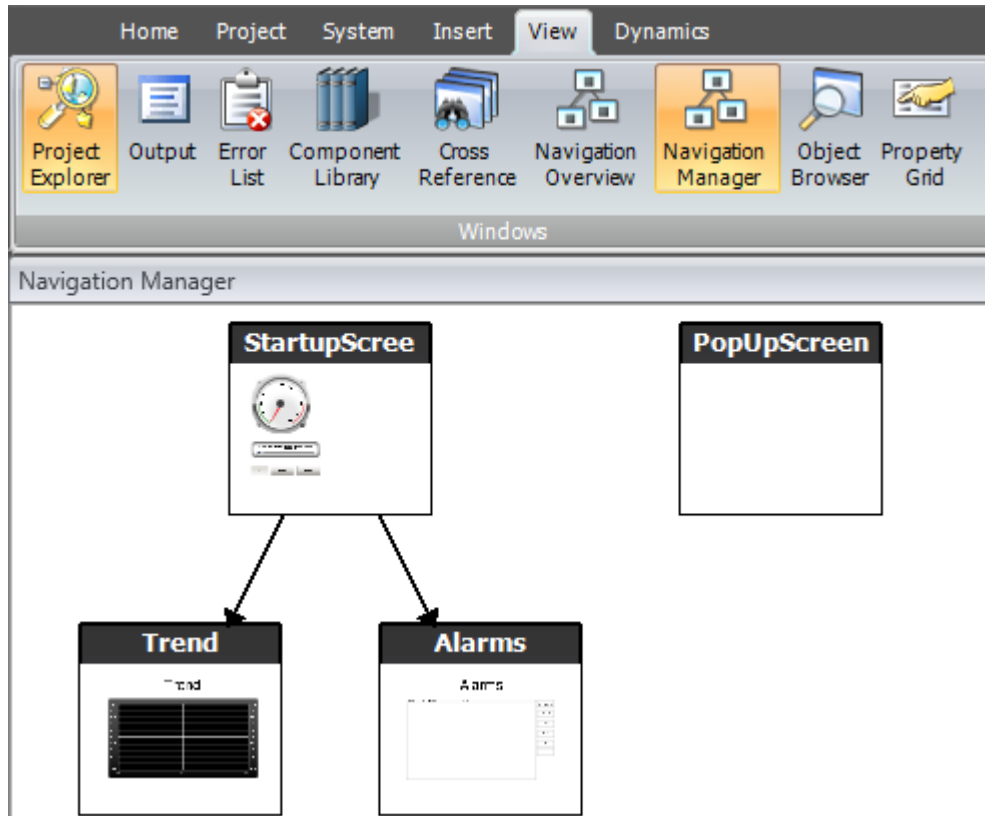
3.5.1 Screen Relations

The Navigation Manager is used to create relationships between screens.

Dragging the mouse pointer from one screen to another screen creates a relation. A button with a **Click** action is defined in the first screen. This **Show Screen** action has the second screen as target, and when clicking it in runtime, the second screen is opened.

Double-click on a screen to open the screen for editing in the desktop area.

3.5.2   Add Screen

New screens can be added directly in the Navigation Manager.

Dragging the mouse pointer from an existing screen to an empty location in the Navigation Manager area creates a new screen, with a relation to the original screen. A button with a **Click** action is defined in the first screen. This **Show Screen** action has the second screen as target, and when clicking it in runtime, the second screen is opened. The button must be removed or changed manually if the new screen is deleted later.

3.5.3   Links in the Navigation Manager

A structure of screens and their relations to other screens will be imposed and shown in the Navigation Manager. The Navigation Manager will provide an overview to help add screens and manage the link relations between project screens.

Right-click anywhere in the Navigation Manager to change link appearance.

Link setting	Description
Show Same Level References	Filters out links to screens on the same hierarchical level.
Show Back References	Shows navigation links to higher hierarchical levels. When not selected, only links to screens on lower levels and links between screens on the same level are shown.
Directed Links	Draws straight link lines between screens, using the shortest way.
Orthogonal Links	Draws link lines in an orthogonal pattern.

Changing the setting or the type affects the appearance in the Navigation Manager only.

3.5.4   Navigation Overview

It is also possible to display the **Navigation Overview** by selecting it from the **View** ribbon tab. It provides an overview of all screens included in the project, with zooming facilities.

3.6   Project Explorer

The Project Explorer shows all screens and components included in the application. It can be activated/deactivated from the View tab.

The Project Explorer is divided into five groups; . **Screens**, **Functions**, **Data Loggers**, **Script Modules** and **Recipes**.
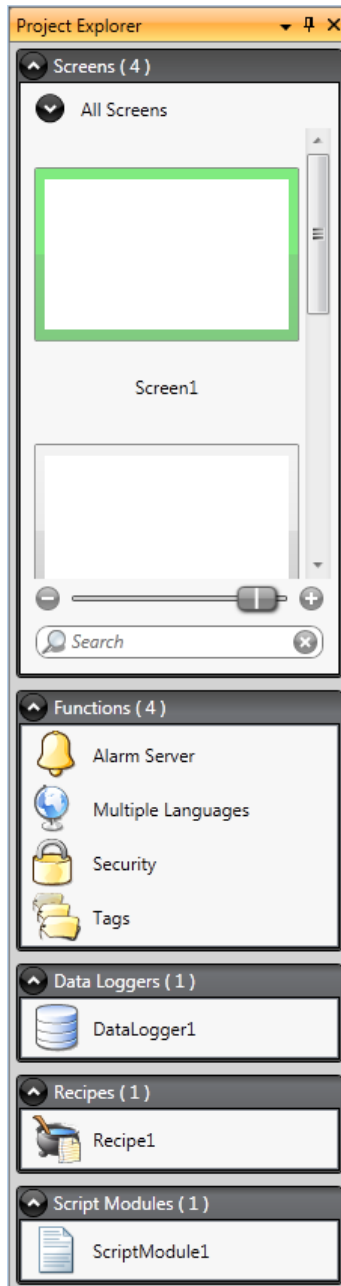
3.6.1 Project Explorer Groups

Click on the group heading to expand/collapse the groups. The number of components in each folder is apparent from the group headings. Right-clicking on a group opens a subset of the commands from the **File menu** and **Insert** tab.

Right-clicking on a component in a Project Explorer group opens a context menu with a selection of common operations. Not all commands are available for all components.

Context menu command	Description
Show	Opens the selected project component for editing in the desktop area
Rename	Renames the selected project component
Add	Adds a new project component to the project
Import	Imports a project component via browsing the hard disk; e.g. a screen from another project
Delete	Deletes the selected component from the project

Screens

The **Screens** explorer lists the screens included in the application. Clicking on a screen opens it for editing in the desktop area.

Screen Context Menu

Right-clicking a screen opens a context menu with a selection of available operations:

- Show
- Rename
- Import
- Delete
- Set as startup screen
- Save screen as template screen
- Add to screen group
- Remove from screen group

The screen set to be the startup screen is marked with a green frame in the **screens** explorer.

Groups

The **Screens** explorer offers a possibility to assign screens to groups.

- Right-click on the **All screens** heading in the **Screens** explorer and select **Add group** followed by entering a unique name for the group in the appearing dialog.
- To assign a screen to a group, simply right-click a screen in the **Screens** explorer and select "Add to Screen Group" from the context menu. Select a group to add the screen to from the appearing "Select Screen group" dialog.
- To show available groups, double-click the **All screens** heading in the **Screens** explorer. Screens not assigned to a group will be listed as **ungrouped screens**.
- Right-clicking a group enables a context menu from which you can choose to rename or delete the group. Deleting a Screen group also deletes all of its assigned screens.

It is also possible to assign a screen to a group by a simple drag and drop operation. Drag the screen symbol and release it on the group.

Zoom Slider

The Zoom slider is found in the **Screens** explorer and provides a way to quickly zoom in and out of your **screens** explorer.

Search

The search function provides the option to search for screens using the screen name or **screen-id**.

The search is only performed within the selected group.

Functions

The Alarm Server, Multiple Languages and Security functions are predefined, as well as Tags, and cannot be deleted from the Functions folder. The other functions can be added from the Insert menu. Most functions will be available from the Functions folder after having added them to the project, but some functions are assigned folders of their own.

Clicking on the function in the Functions folder opens its configuration pages for editing in the desktop area.

Alarm Distributor

Alarms can be distributed between operator panels, with notification via printer, SMS or e-mail using the alarm distributor.

Related information
Alarm Distributor

Alarm Server

The Alarm Server handles alarm groups and alarm items.

Related information
Alarm Server
Alarm Groups
Alarm Items

Audit Trail

The Audit Trail function enables tracking of operator actions.

Related information
Audit Trail

Function Keys

Function keys for operator panels as well as for the regular PC keyboard can be configured.

Related information
Function Keys

Multiple Languages

Configuring multiple languages for user texts and system texts are supported. The texts can be exported, edited and reimported to the project. A tool for automatic translations is also included.

Related information
Language Management

Reports

The reports function allows adding excel report templates to the project.

Related information
Reports

Scheduler

A scheduler can be used to control events in the process at special times.

Related information
Scheduler

Security

Users and groups of users can be set up for security purposes.

Related information
Security Management

Tags

Clicking on **Tags** in the Functions folder displays the tags configuration page in the desktop with all the tags that are used in the application. Tags are added by clicking on the **Add Tag/Add System Tag** button.

Tags can belong to a connected controller, or be internal. Internal tags can be used and selected in the same way as a regular controller tags.

System tags are used to monitor or access variables from the system, for example *Used RAM* or *DateTime*. System tags are indicated by blue color in the tag list.

Related information
Tags
Internal Tags
System Tags

Text Library

With the text library function, text tables can be created, where values are linked to texts.

Related information
Text Library

Recipes

Several recipes can be inserted in the project. Each recipe function handles a pre-defined set of recipe items.

Related information
Recipe Management

Script Modules

Clicking on a script module in the Script Modules folder opens the script editor in the desktop area. A script module can be included to program features not covered by included functions and actions, or to be used to share functionality among functions.

Related information
Script

Data Loggers

Data can be logged and saved to a database on time intervals or depending on changed values.

Related information
Data Logger

3.7   Component Library

The Component Library contains reusable components for a project.

3.7.1   Components

The Component Library includes a large number of predefined graphical objects, categorized into groups. User-defined graphical objects and other files may also be saved in the component library. All components in a sub folder must have unique names.

Examples of components that may be included in the library:

- predefined components
- objects
- text objects configured with multiple texts
- symbol objects configured with multiple symbols
- grouped objects
- external pictures
- text
- folders (file folders)
- htm/html files
- media files
- PDF files

Since plain text can be saved as a component, it is possible to save reusable script components by dragging a section of script code from the Script view into the Component Library.

A component is displayed with its name and a thumbnail representation in the Component Library window.

Grouped objects are displayed as a thumbnail representing the primary selection of the object. Text files and folders are represented by operating system thumbnails.

Project Pictures

The Project Pictures folder is project specific, and contains pictures that are used in the current project. The components in Project Pictures are stored in the project's Symbols folder.

Project Files

The Project Files folder is project specific, and contains files that have been added when creating the project.

Project files and folders can be added and deleted from the Project Files folder before downloading the project to an operator panel. The Project Files folder is located in the Project folder.

For instance, "C:\MyProjects\ProjectFolder\Project Files".

It is also possible to access the operator panel via ftp to add and delete files and folders after the download, provided that the FTP service is enabled in the project.

Related information
FTP
Download

3.7.2   Add and Use Components

Drag-and-drop any component from the library to the current screen, or use copy [**Ctrl + C**] and paste [**Ctrl + V**] commands.

Objects can be copied or moved to the Component Library.

A text file, that is saved in the Component Library will be converted to a text box object when pasted into a screen.

A component (picture or group) that is originally copied from an object with a tag connection will keep its tag connection and any dynamic properties when used from the Component Library.

Note:
If a component with a tag association is reused in another project then all referenced tags must be present in the new project.

3.7.3 Component Library Window

The component library is available as a tab at the right-hand side of the desktop. The window is configured with auto-hide functions, and can also be activated/deactivated from the View ribbon tab. Its appearance can be managed just as other tools; as docked or floating windows.

User-Defined Categories

Right-click on an empty space of the Component Library window and select **New** to create a new category, and select a name of your preference. User-defined categories are not project specific. This means that all objects saved in a user-defined category will be available in all iX Developer projects.

Expand and Collapse Folders

The top area is a contents frame that is used to navigate through the folders of the Component Library. The contents heading displays the name of the currently selected folder. The folder view can be expanded and collapsed with the control to the left of the displayed folder name.

Search

The **Search** feature provides a case-insensitive and alphabetical search, based on the component label, in the current folder and in all sub folders.

Zoom

The Zoom slider at the bottom area of the tool window can be used to change the size of the displayed components. Resting the mouse pointer over a component displays a full-size picture of the component.
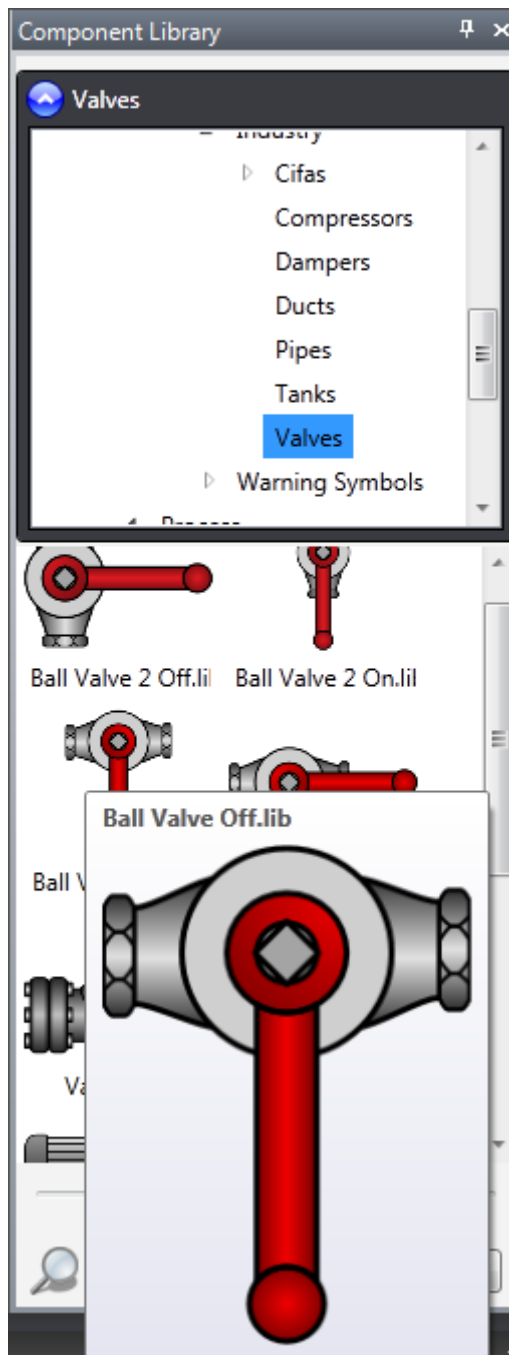
Component Library window, displaying full size of a selected component

3.8 Property Grid

Most properties and settings for an object or screen are defined from its General and Home tabs.

A detailed overview of object properties is available by selecting **Property Grid** from the View ribbon tab or by selecting **Properties** from the context menu.

The property grid includes more properties and settings than the ribbon tabs, for example size and position of objects, and delay of mouse and touch screen input in runtime.

Property groups can be collapsed/expanded.

The window is configured with auto-hide functions. Its appearance can be managed just as other tools; as docked or floating windows.

3.8.1   Search

The Search feature provides a case-insensitive and alphabetical search, that makes it easy to reduce the number of displayed properties.

3.9   Object Browser

An overview of all objects included in a screen can be displayed in the Object Browser.
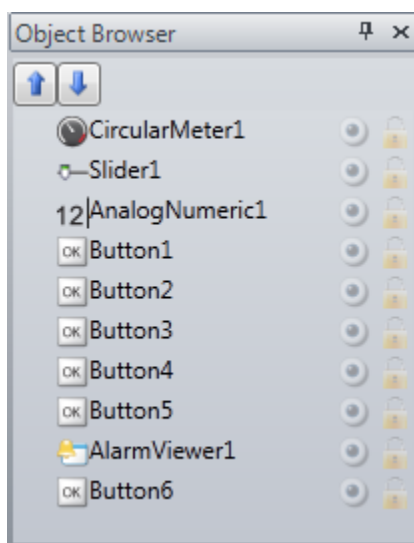
It is easy to bring objects that may have been obscured by other objects up front, by using the arrows at the top left.

Clicking the buttons at the right side of each object enables simply hiding/showing and locking/unlocking objects.

Note that the top-to-bottom order in the Object browser corresponds to the front-to-back order on the screen.

3.10   Output

The Output window contains messages concerning the project validation and build. Created project files with folder paths and any found errors will be listed in Output.

Note:
Incorrect code entered in text mode (Xaml and Script) may result in errors that are not possible to recognize during analysis and build.

3.11 Error List

The Error List contains information about problems and errors detected in iX Developer for the current project. Build errors are reported in the Output window.

3.12 Help

The Help file for iX Developer can be opened by pressing **F1** on the keyboard or using the **Help** button in the ribbon tab heading.

4   Tags

Objects connected to tags can change values in a controller, and controller values can be reflected by changing object appearance in various ways. Objects in a screen will remain static until connected to a tag.

A tag has a symbolic name and can be of different data types.

Tags can belong to a connected controller, be internal or belong to the system. Internal tags are used to handle local data values, and can be viewed and treated in screens just like regular controller tags.

The global tag list provides an overview of all used tags, regardless of controller. This makes it easy to develop projects that can be used with a variety of controllers. It is also possible to connect a tag to more than one controller simultaneously.

4.1   Adding Tags

Tags are handled by clicking on **Tags** in the Functions folder in the Project Explorer.

Clicking the **Add Tag/Add System Tag** button adds a new tag to the list. Make settings for each added tag. When predefined values are available, these can be selected by clicking the arrow at the right of the table cell.

4.1.1 Basic Settings

To reduce complexity in the tags configuration page, some columns can be made hidden/visible. The basic settings are always displayed. It is possible to customize which columns to show by default using **Tags Editor Options**.

Enter name, data type and access rights for the tag. Select if the tag is to belong to a controller entered on the Controllers tab. If no controller tag address is entered, the tag will be treated as an internal tag.

Parameter	Description
Name	Symbolic name. The name can be any alphanumeric string, beginning with a letter (a-z, A-Z).
Data Type (Tag)	Selection of presentation format; for example used to show the correct engineering unit when using scaling. DEFAULT follows selection made for Data Type (Controller).
Access Rights	Defines access rights to the tag
Controller	Controller address
Data Type (Controller)	Selection of data type

Related information
Tags Editor Options
Scaling

4.1.2 Scaling

Scaling affects only tags connected to a controller.

Checking the **Scaling** box displays the **Offset** and **Gain**

columns. These parameters are used to change the scale of the controller value to a value shown in accordance with the following equation:

Panel value = Offset + (Gain * Register value)

When the value for an object is changed from the operator panel, the scale is changed to the value shown in accordance with the following equation:

Register value = (Panel value - Offset)/Gain

Scaling does not affect the defined maximum and minimum values, or the number of decimals.

4.1.3 Data Exchange

Checking the **Data Exchange** box displays the **Direction** and **When** columns. Data exchange supports real time data exchange between different controllers; also of different brands. The data exchange can be performed when a digital tag is set or at a certain time interval, set up on the Triggers tab. Data exchange can be selected for individual tags, array tags or for the complete range of tags.

Related information
Data Exchange

4.1.4   Others

Checking the Others box makes it possible to configure a number of additional tag properties.

Parameter	Description
Description	An optional description of the tag. Description of system tags is added automatically.
Poll Group	Makes it possible to poll groups of tags at different intervals.
Always Active	Prevents the tag from becoming inactive. This may be useful for actions and scripting purposes. Note: The Active setting for the controller on the Controllers tab is predominant over the Always Active setting for a particular tag.
Non Volatile	Keeps the last tag value in memory even when the panel or PC is powered off. Only available for internal tags.
Initial Value	Makes it possible to assign a tag an initial value at system startup. Only available for internal tags.
Index Register	Makes it possible to select a register from which an object is to collect the shown value.
Action	Makes it possible to configure actions based on a condition of a tag.
Log to Audit Trail	Logs operator actions on the tag. Only available if the Audit Trail function is activated in the project.
Audit Trail Description	Optional static or dynamic descriptions for operator tag actions, that are saved in the Audit Trail database and visible in the Audit Trail Viewer. If **Dynamic Description** is selected, value before change can be included in the description by using {0}; value after change by using {1}. Only available if the Audit Trail function is activated in the project.
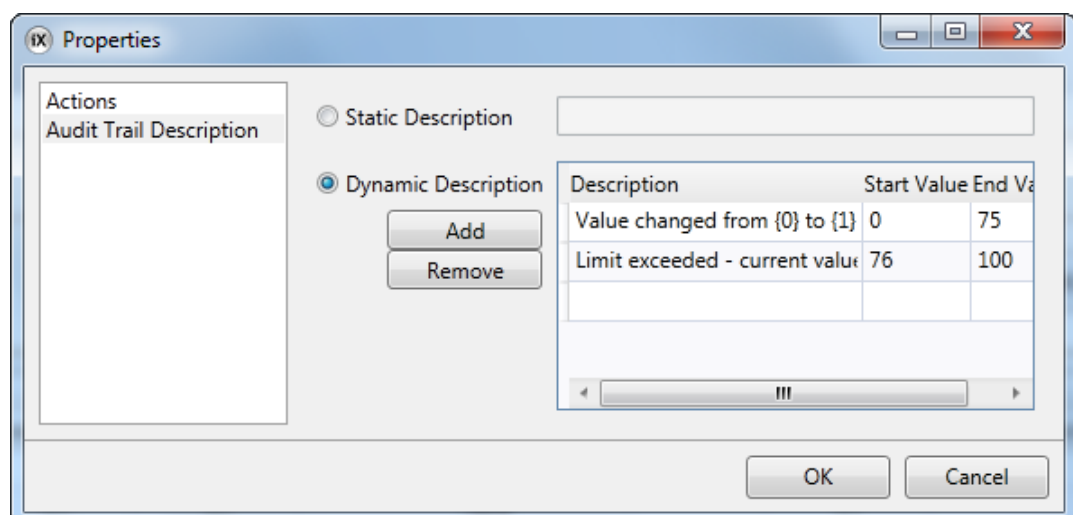
4.1.5   Adding Tags during Editing

Tags can be added directly when editing objects. When an object is selected, the tag association can be managed from the Tag/Security group in the Home or General ribbon tabs, or by right-clicking on the object and using the Mini Toolbar. In both cases, tags are added by clicking the **Add** button.

Note:
Tags added during editing will stay internal until connected to a controller on the tags configuration page.

Related information
Tag/Security Group
Mini Toolbar Options

4.1.6   Selecting Multiple Tags

Multiple rows of tags can be selected using the [**Ctrl**] or [**Shift**] keys. This makes it easy to, for example, delete a number of tags quickly.

4.2 Tag Actions

One or multiple actions can be configured depending on a changed value of a tag. The Action column is displayed when the **Others** box is checked in the Tags configuration page.

Clicking **...** in the Actions column for the selected tag brings up the Action properties window. First, a condition is configured by clicking **Add** under Condition. Select a condition, enter a value and click **OK**.
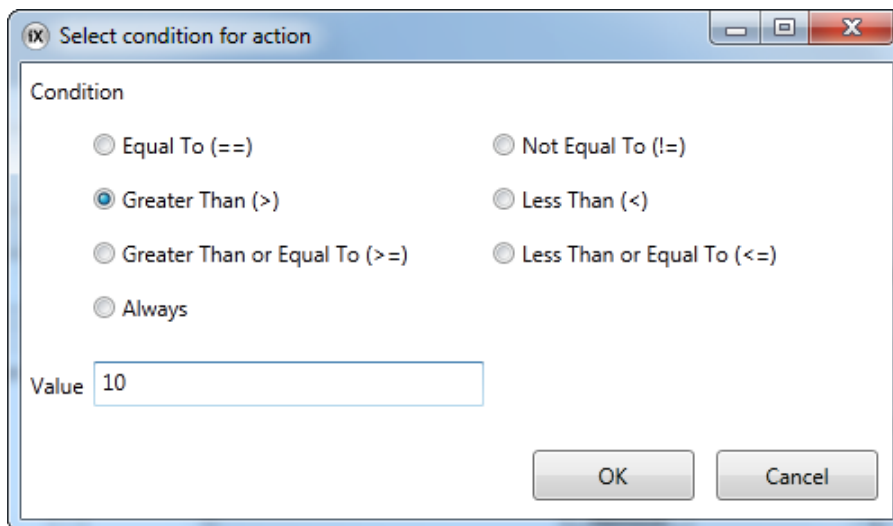
Then, configure actions to be executed when the condition is met.
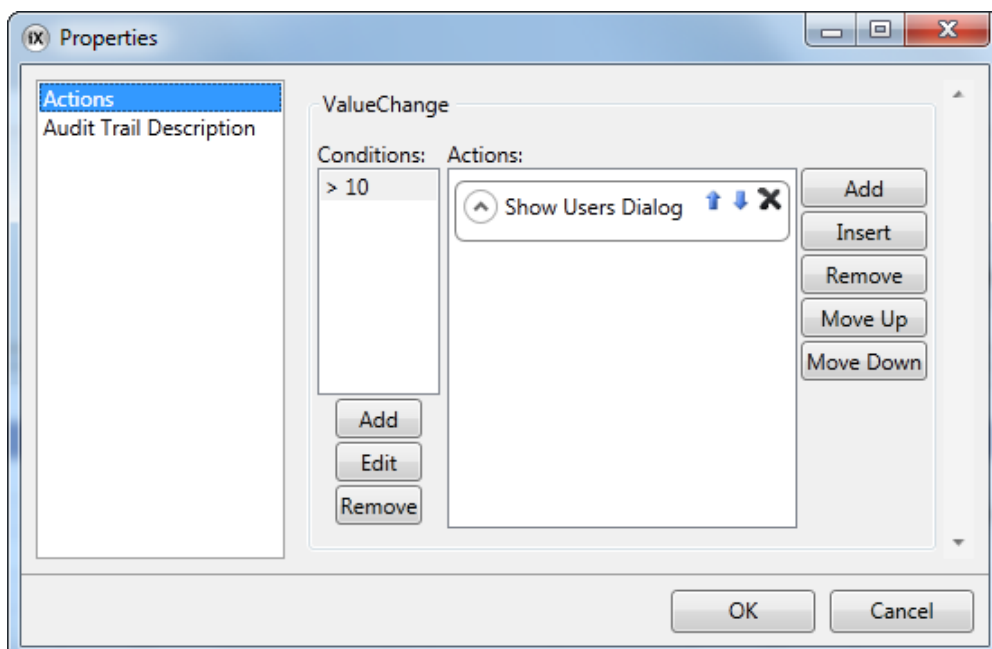
Related information
Actions Ribbon Tab

4.3 Internal Tags

Internal tags can be used to calculate values that need not be represented in the controller, for example information only for the operator. Internal tags are selected in the same way as an external controller tags. Any number of volatile user defined tags can be created.

Note:
Internal tag values will not be saved in the operator panel when the project is restarted, unless the **Non Volatile** option is used.

Unlike the regular drivers, it does not matter if internal tags are used in consecutive order or not.

Note:
The **ValueOff** and **ValueOn** events for internal variables are available for selection in script mode, but they are not executed when running the project.

4.4   System Tags

System tags are used to monitor or access variables from the system. System tags are handled as regular controller tags, i.e. can be used in screens, logged, included in scripts etc. The Date and Time system tags can be used when you want to use the controller clock in the panel project. The following system tags are available:

System tag group	System tag	Description
Communi-cation	Communication Error Message	Latest communication error message
	Communication Errors	Number of active communication errors on all controllers
	Remote Alarm Server Connection Errors	Number of remote alarm servers with connection errors, e.g. disconnected servers.
Date and Time	DateTime	Current date and time
	Day	Day component of current date
	Day of Week	Day of current week (1-7 where 1 is Sunday)
	Hour	Hour component of current time
	Minute	Minute component of current time
	Month	Month component of current date
	Second	Second component of current time
	Year	Year component of current date
Project	Current Screen Id	The ID of the current screen. The data type of this tag is int16 and it only has read access. This tag gets updated every time you open a screen, with the exception on pop up screens. When opening a pop up screen the current screen id tag will not be updated. When entering a screen without a screen id the tag value of the current screen id will be set to -1. Note that 0 is a valid screen id.
	Current Screen Name	Name of the current screen
	Latest Loaded Recipe	Name of the latest loaded recipe
	New Screen Id	The ID of the new screen. The data type of this tag is int16 and it has read/write access. The tag is always active. When the new screen id tag changes value a screen jump is made to the screen with corresponding screen id. If no such screen exists a notification message will show. If the new screen contains configured screen instances (alias), the default instance is always displayed.

System tag group	System tag	Description
System	Project Name	Name of the current project
	Runtime Version	Runtime version of framework
	Screen Update Time	Screen update time (in ms)
	Available RAM	Amount of available RAM memory (in kB)
	Available Storage	Amount of available storage memory (in MB)
	Backlight Brightness Level	Backlight brightness level (in percent)
	CPU Load	Amount of used CPU capacity (in percent)
	Current User	The user name of the currently logged in user
	Debug Logging Enabled	Indicates if debug logging is enabled.
	Internal Temperature	Internal panel temperature (in °C)
	Physical RAM	RAM memory installed in computer (in kB)
	Power LED Blink Frequency	Blink frequency of the power LED (1–10 Hz, 0 = always on)
	Power LED Blue Intensity	Intensity of the blue power LED (0–100)
	Power LED Red Intensity	Intensity of the red power LED (0–100)
	Storage Memory	Storage memory installed in computer (in MB)
	Used RAM	Amount of used RAM memory (in kB)
	Used RAM Percent	Amount of used RAM memory (in percent of physical)
	Used Storage	Amount of used storage memory (in MB)
	Used Storage Percent	Amount of used storage memory (in percent of all storage)
	Windows CE Image Version	Windows CE Image Version

System tags are indicated by blue color in the tag list.

System tags are added by clicking the **Add Tag/Add System Tag** button in the tags configuration page.

4.5   Array Tags

An array tag is a tag containing multiple values.

If the array size is >1, the tag is considered as an array tag.

Note:
Array tags are only supported by controllers which use addresses that can be numerically increased.

4.5.1   Array Tag Set Up

Clicking the **Add** button adds a tag to the list. Make settings for each added tag. When predefined values are available, these can be selected by clicking the arrow at the right of the table cell.

Clicking the **Data Type** cell in the **Controller** column brings up the data type configuration page. An array tag is configured by setting the array size to a value >1.
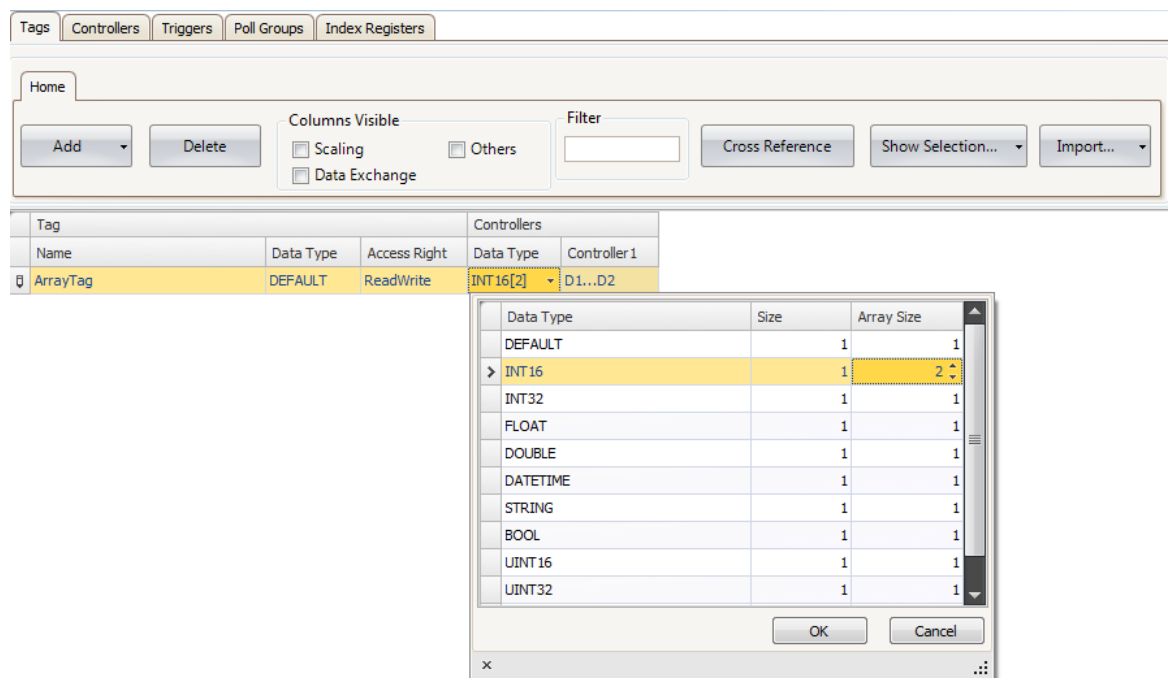
Data type	Description
Data Type	Selection of presentation format.
Size	
Array Size	Set the array size. If the array size is >1, the tag is considered as an array tag.

Limitation

When using controllers with named variables and accessing individual bits with the syntax VariableName.BitNo, array tags of data type BOOL with bit addressing, are only supported for 16-bits tags.

Related information
Tags
Chart Object
Data Exchange

4.6   Cross Reference

The Cross Reference tool provides an overview of where a specific tag is used in the current project.
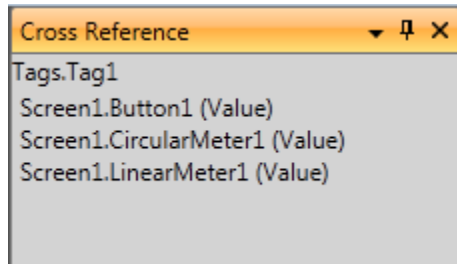
All occurrences of the tag in the current project are listed. Double-clicking on an occurrence in the list automatically brings you to the object or function where the tag is used.

The cross reference tool can be accessed by clicking the cross reference button on the **Tags** tab or on the View ribbon. It can also be accessed from the context menu.

4.7 Triggers
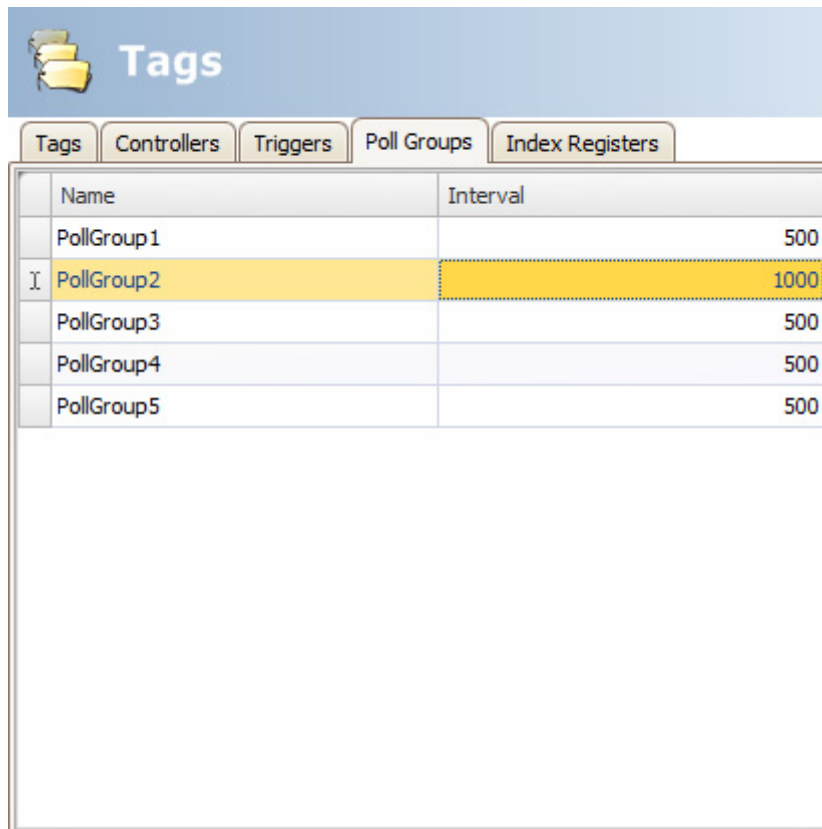
Triggers are used to determine when data exchange is to be performed; based on tag or a certain interval.
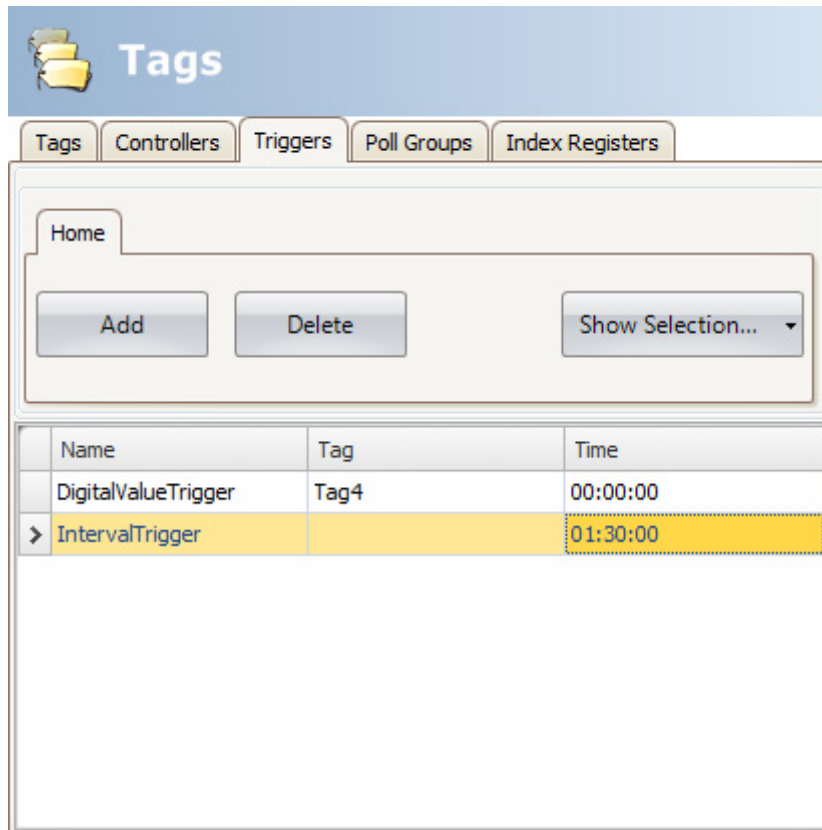
Parameter	Description
Name	Symbolic name. The name can be any alphanumeric string, beginning with a letter (a-z, A-Z).
Tag	Selection of a tag, which when changed to a value other than 0, triggers data exchange
Time	Time interval between data exchange operations in the format HH:MM:SS

4.8 Poll Groups

Poll groups allow grouping tags for polling at 5 different intervals. The groups defined on the Poll Groups tab are available for selection from the Tags tab after checking the **Others** box.

Parameter	Description
Name	Symbolic name. The name can be any alphanumeric string, beginning with a letter (a-z, A-Z).
Interval	Polling interval in ms

4.9 Station Handling

For some controllers, it is possible for the operator to define from which controller station to read or write values. The station number is given as a prefix to the controller address.

The station number for each controller is defined on the Controllers tab of the Tags configuration page, by clicking the **Settings** button.

The default station is set on the **Settings** tab, and the stations are defined on the **Stations** tab.
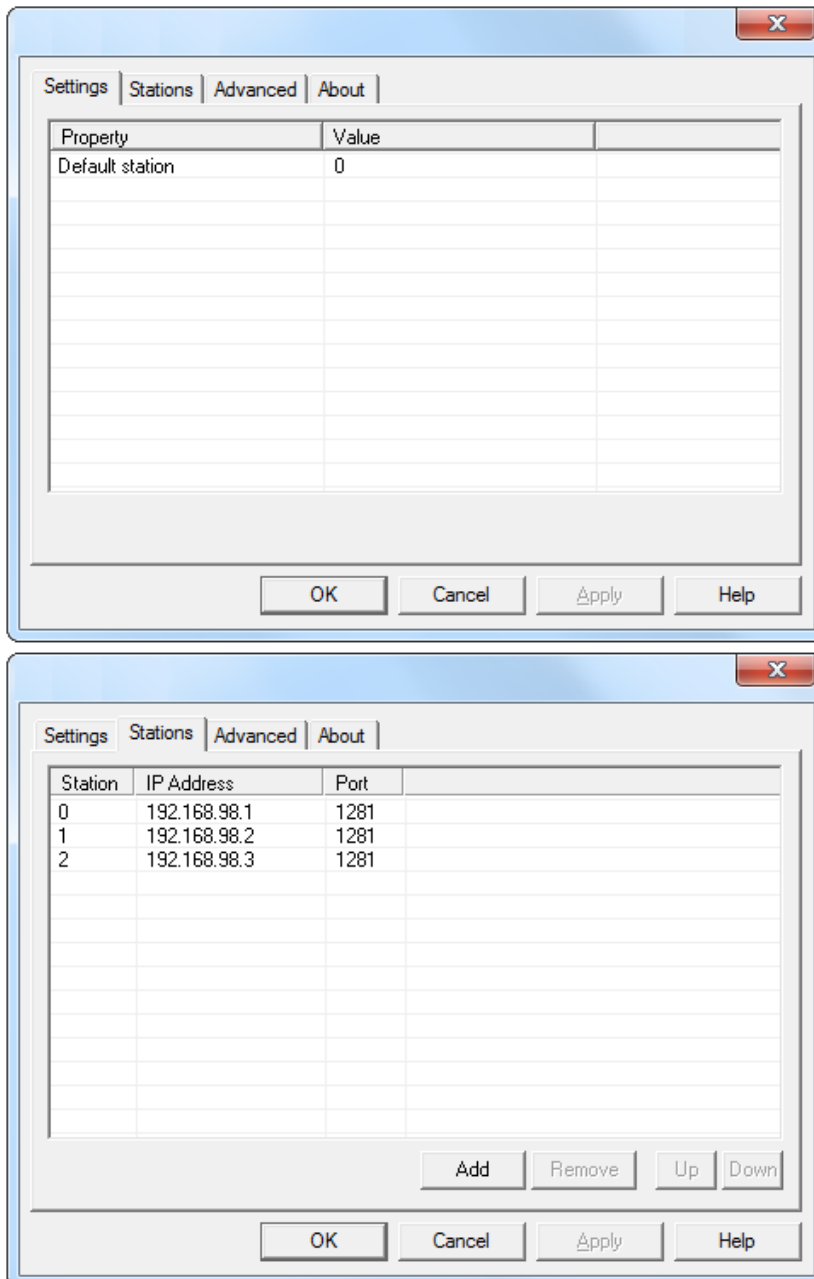
In this example, from an Ethernet driver, Station 0 is the default station.

The default station is the station addressed without using a prefix. The other stations can be addressed either as a fixed number or as an index register. With fixed station numbers, an object is always connected to the same controller address, and only the value of this controller address can be used in the object.

When addressing for example D10 in station 2, the following syntax is used: *2:D10*

Related information
Using Index Registers for Station Handling

4.10 Index Registers

With index addressing, it is possible for the operator to select from which tag an object is to collect the shown value. Without index addressing an object is always connected to the same controller address, and only the value of this address can be shown in the object.
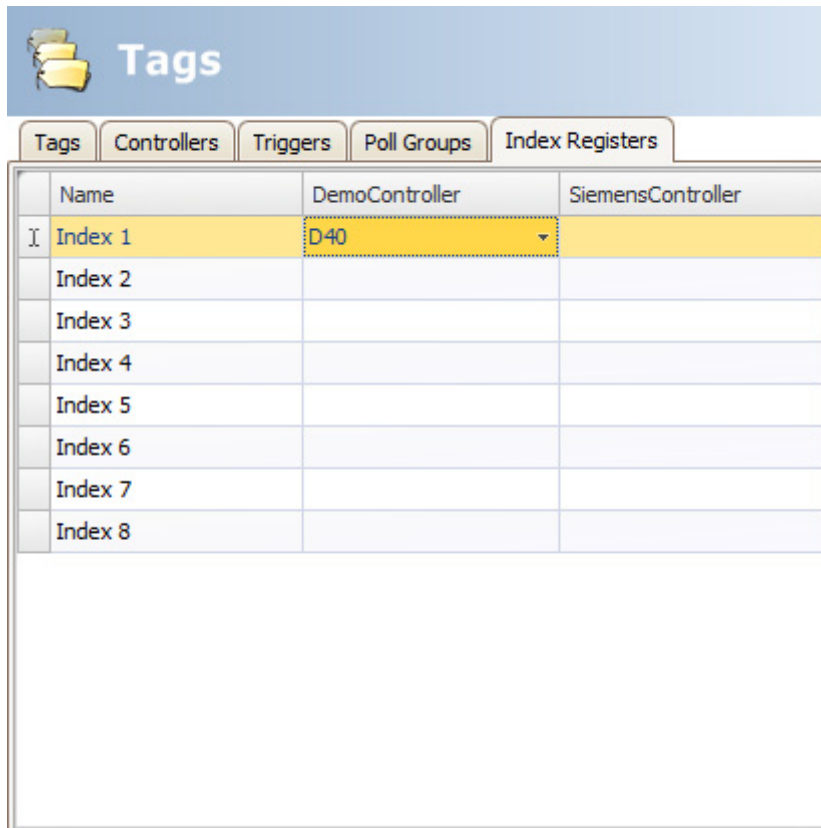
Which tags to use for indexing are defined on the Index Registers tab of the tags configuration page. Up to eight different index registers can be used for each controller, and each index register can be used for more than one object. Index registers can be connected to any tag in the global tag list, but the tag has to be selected for each controller.

Tags are connected to index registers on the Tags tab.

In general index addressing can be formulated as:

Display value = the content in tag (the object's address + the content in the index register)

If the content in the index register is 2, and the address of the tag specified in the object is 100, the value shown in the object will be collected from address 102. If the value in the index register is changed to 3, the value in the object will be collected from address 103 instead.

4.10.1   Index Addressing Example

Using index registers may be convenient when for example controlling torque and speed of a number of connected motors from the same screen, without having to configure and display maneuverable objects for each motor.

The following example is based upon the presumption of three motors, and the torque and speed of each motor is held in a different tag:

Motor 1		Motor 2		Motor 3	
Torque	**Speed**	**Torque**	**Speed**	**Torque**	**Speed**
D11	D21	D12	D22	D13	D23

1. Create a project where the Demo Controller is selected as controller.
2. Add tags and labels and connect them to Demo Controller tags according to below:

3. Select the Index Registers tab and connect **Index 1** to the **MotorIndex** tag (D0).
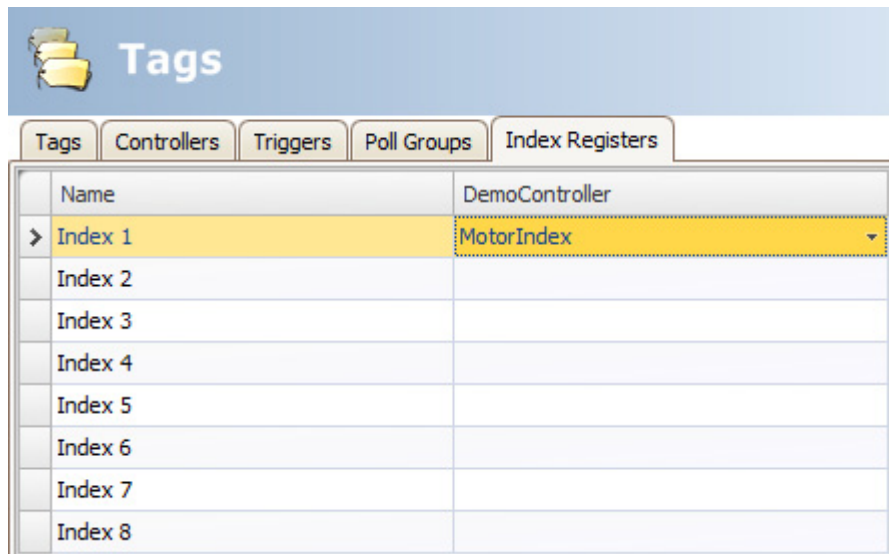
4. Select the Tags tab again and check the **Others** box in order to display the Index Registers column.

5. Connect the **Torque** and **Speed** tags to Index Register 1.
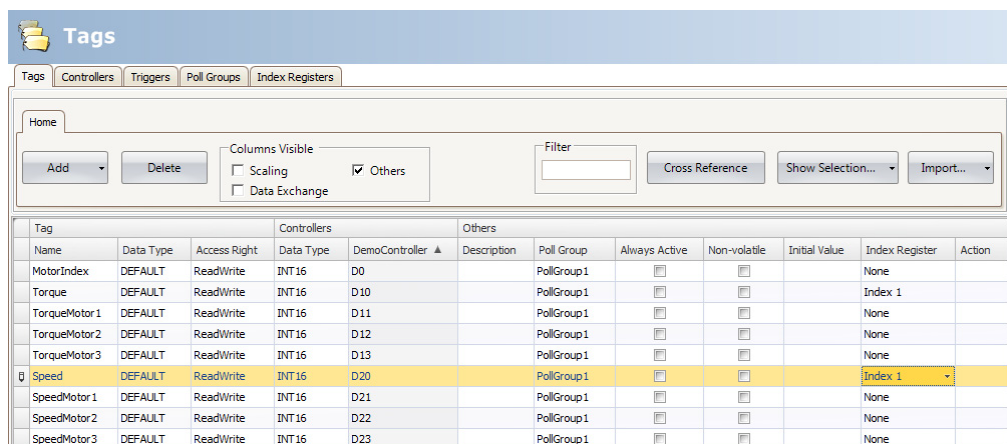
6. Now, create a ComboBox object for selection of motor, and two analog numeric objects that will display (or set) the Torque and Speed tags for each motor. Also add explaining labels.

7. Click **Configure Texts** for the ComboBox object, connect the object to the MotorIndex tag, and add three texts and values according to below.
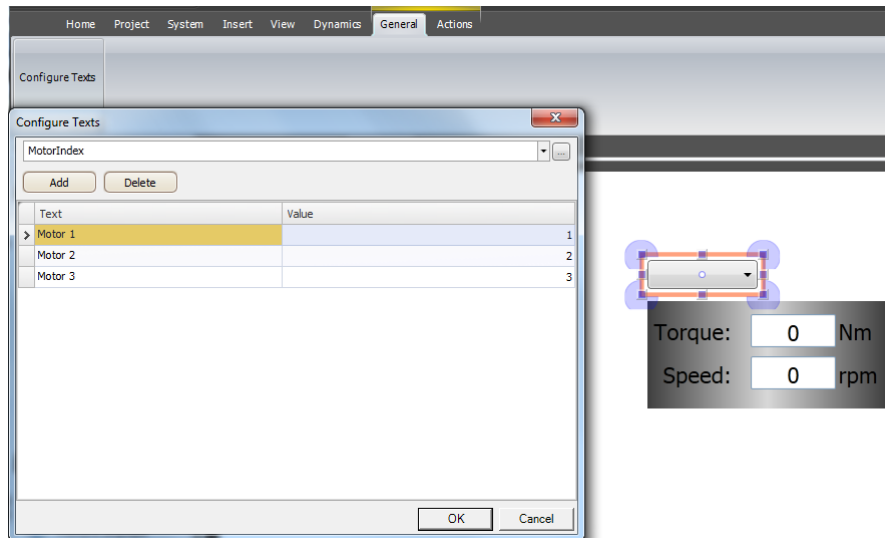
8. Connect the analog numeric objects to the Torque and Speed tags.

9. In order to test that the index addressing works even though no controller is connected, add maneuverable objects, e.g. sliders, and connect them to the D11–D13 and D21–D23 tags.

10. Run the project, enter values for the objects connected to D11–D13 and D21–D23, and select the different motors using the ComboBox.
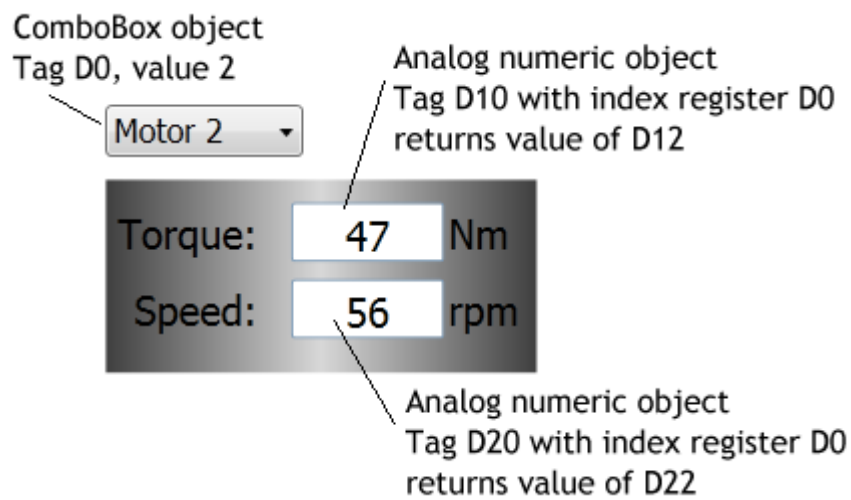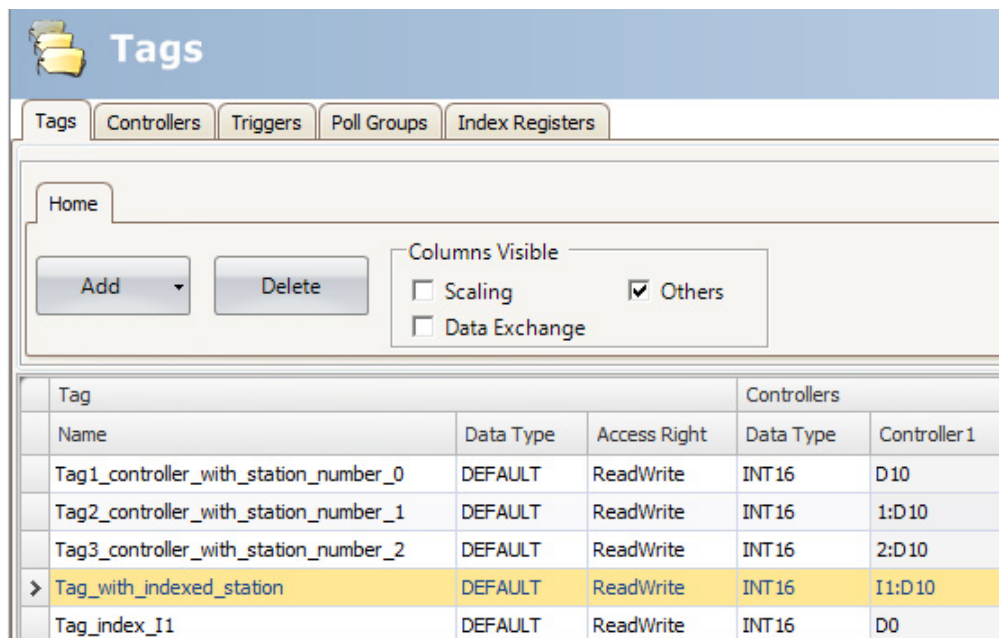
4.10.2 Using Index Registers for Station Handling

For some controllers, it is possible for the operator to define from which controller station to read or write values. The station number is given as a prefix to the controller address.

This is stated either as a fixed number or as an index register. The default station is the station addressed without using a prefix. The other stations can be addressed either as a fixed number or as an index register. With fixed station numbers, an object is always connected to the same controller address, and only the value of this controller address can be used in the object.

Index addressing reduces the number of tags needed as well as the number of objects to be configured in the iX Developer project when showing values from / writing to multiple controller stations.

The syntax when using index registers for station handling is for example *I1:D10*.

This means that the value in the controller address connected to Index 1 will point out which controller station number to read or write values.
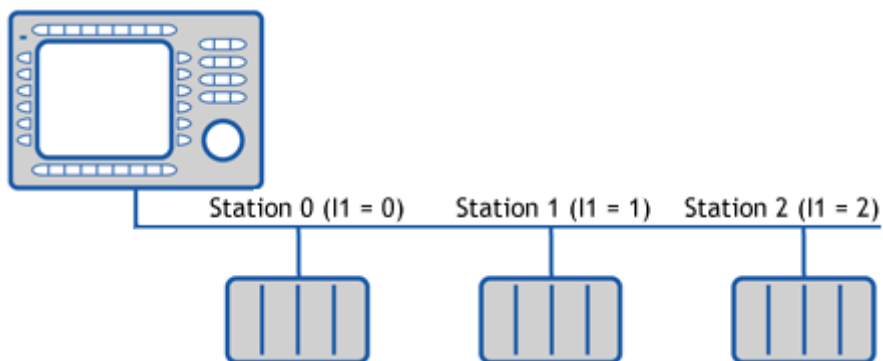
Related information
Station Handling

4.11 Expressions

The Expressions functionality is used when connecting an object to a tag and the purpose is not to use the tag value as it is. Expressions enable the user to e.g. extract a bit from a tag, add a dynamic offset or gain, and then to use this value for a specific object.

Note:
Static offset and gain for a tag is set in the tags configuration page.

The same expression can be connected to more than one object in the project.

4.11.1   Definition

An expression is a one line return statement in C#. The default expression code is "value".

4.11.2   Using Expressions

Click the button next to **Select Tag** on the **General** ribbon tab when an object is selected. The Expressions dialog is displayed.
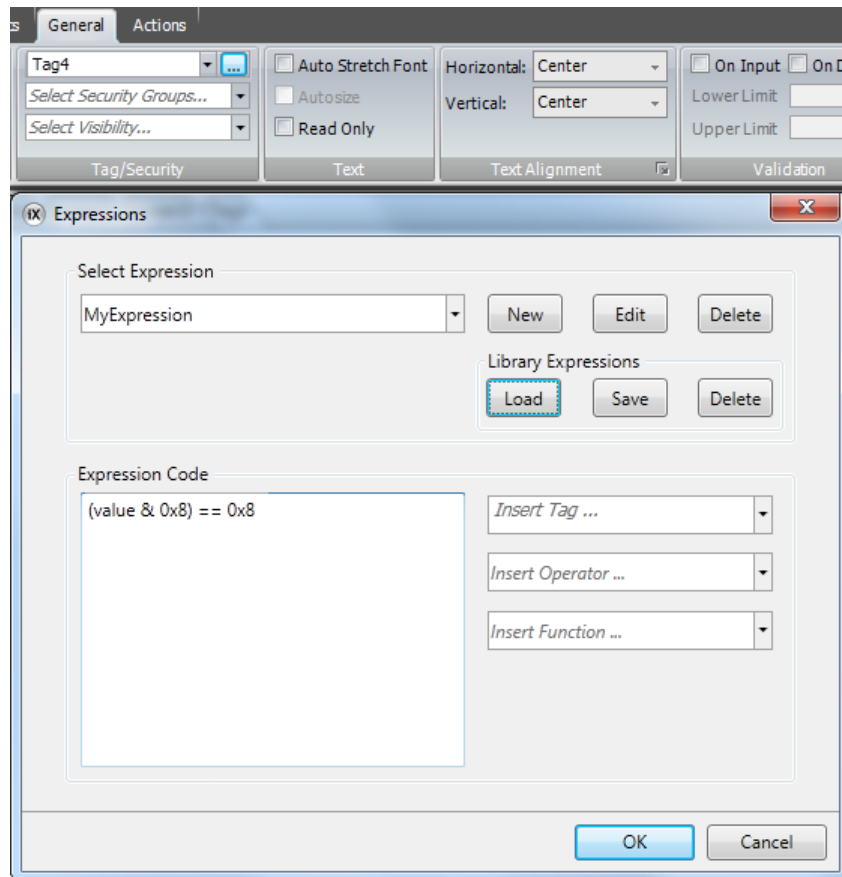
Click **New** to create an expression and expression group (optional). Existing expressions are available from the **Select Expression** drop-down menu. Use the **Edit** and **Delete** buttons to edit or delete existing expressions.

Select among the tags, operators and functions to insert. It is also possible to type the expression manually, but without assistance from IntelliSense (auto completion). If an expression is not valid, an error message is given when validating the project. The expression is run, when the value of the original tag that is connected to the object is changed.

Expressions added to an object in a project are available only in the current project. If an object with a local expression is saved into the Component Library, the object can be used in an other project, but the expression is not included. In order to reuse expressions, they have to be saved in the expressions library.

Related information
Library Expressions

4.11.3 Library Expressions

Examples of expressions (how to extract bits from an int) are included when installing iX Developer and can be added to the project by clicking **Load**.

Expressions can be reused in other projects by saving them in the expression library. Select the expression in the drop-down list and click **Save**. The expression is saved to a file in the PC. To load the expression in another project, click **Load**. A selection dialog is displayed. Select the expression in the drop-down list and click **OK**.

If you have saved expressions to the library that you want to remove, click **Delete** and select among the saved expressions in a dialog. Expressions included when installing iX Developer are not listed, since these cannot be deleted.

4.11.4 Limitations

An expression is executed only on *read*, i.e. when the value is changed in the controller.

When writing a value (from e.g. an analog numeric tag), the expression is not executed.

If another tag than the original tag is used in the expression, the expression is not triggered on its value change.

If an expression that is used in more than one location is either renamed or deleted, references will not be updated. A build error will be given.

Expressions can be used for all data connections except for

- actions
- the Property grid
- recipe item tags
- current language tag in Multiple languages configuration
- trigger tags
- index tags
- scheduler tags
- dynamic text for alarm items

4.12 Data Exchange

Data exchange between different controllers is supported; also between controllers of different brands. The data exchange can be performed when a digital tag is set, or at a certain time interval. Data exchange can be selected for individual tags, array tags or for the complete range of tags.

Note:
The data exchange function is isolated from general tag/controller settings. For example, a tag assigned read access only for the operator, may still be read and written by the data exchange function.

Follow the steps below to set up data exchange with three different controllers involved:

1. Select the **Controllers** tab of the tags configuration page and enter short names for the controllers in the ID column, and make sure that all controllers are active.
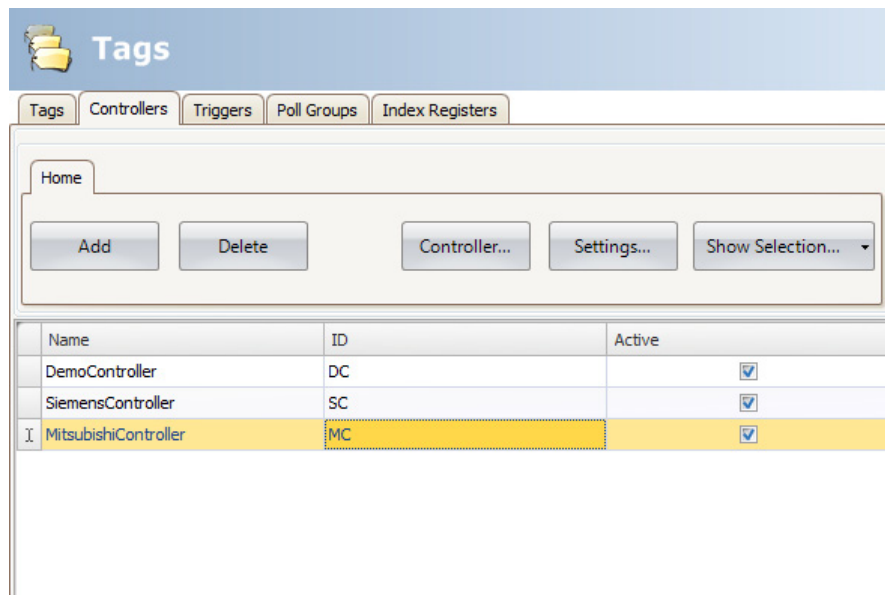
2. Select the **Tags** tab and check **Data Exchange** to display the Direction and When columns.

3. Select an address that is represented in all of the controllers.

If an array tag is selected, the complete address range corresponding to the array size will be exchanged:

4. Click **...** in the Direction column to set up the direction of the data exchange.
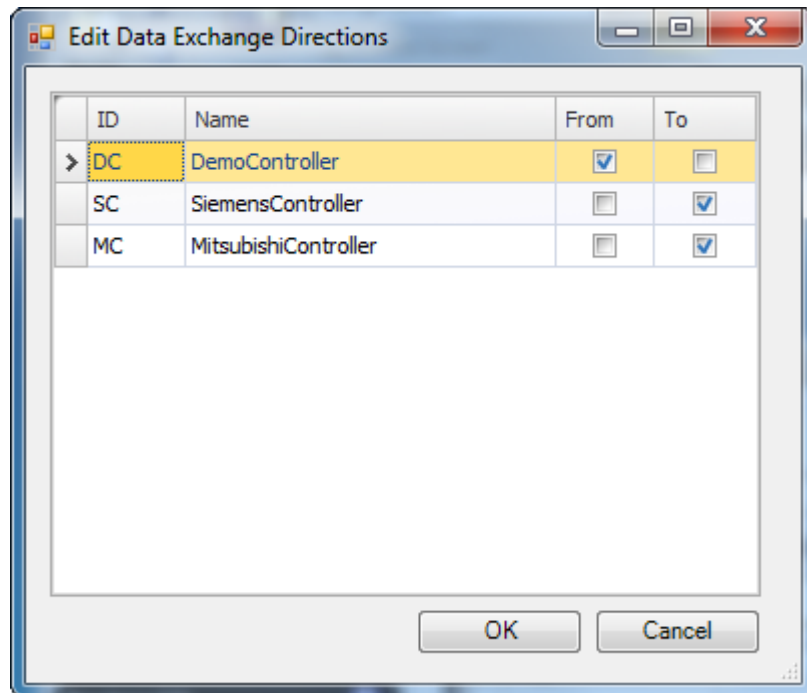
5. Add a trigger tag on the Triggers tab with time interval for data exchange or use a tag, which when greater than 0, will trigger the data exchange. Alternatively, set up a trigger that includes both value change and a time interval.

6. Return to the Tags tab and select which trigger to use.

Tag			Controllers				Data Exchange	
Name	Data Type	Access Right	Data Type	DemoCo...	Siemens...	Mitsubishi...	Direction	When
Tag1	DEFAULT	ReadWrite	INT16	D0	V10			Value Change
Tag2	DEFAULT	ReadWrite	INT16	D1	V11	D3		Value Change
Weight	DEFAULT	ReadWrite	INT16	D10	V12			Value Change
Length	DEFAULT	ReadWrite	INT16	D11	V13			Value Change
Width	DEFAULT	ReadWrite	INT16	D12	V14	D4	DC -> SC, MC	DataTrigger2▾
SystemTagDateTime	DEFAULT	Read	DATETIME					Value Change
Tag5	DEFAULT	ReadWrite	INT16		V20	D5		Value Change
Tag6	DEFAULT	ReadWrite	INT16		V22	D6		Value Change
Tag7	DEFAULT	ReadWrite	DEFAULT					Value Change
SystemTagAvailab...	DEFAULT	Read	INT32					Value Change
Tag4	DEFAULT	ReadWrite	BIT	M0				Value Change
D40	DEFAULT	ReadWrite	INT16	D40				Value Change
Counter	DEFAULT	ReadWrite	INT16	C0				Value Change

Related information
Triggers

4.13 Importing and Exporting Tags

Tags can be imported to and exported from an iX Developer project, to, for example, be edited in Excel or a text editor, or to be transferred to another project. The complete tag list may be imported and exported, or the actions can be based on connected controllers.

The import function also allows selecting individual tags for import.

4.13.1 Handling Columns

The import function allows skipping columns, or binding columns in the import file to any column in the Tags configuration page.

The **Name** and **DataType** columns must always be bound.

A column may only be bound once.

4.13.2 Saving the Import Configuration

For repetitive import actions the import configuration, e.g. how columns are bound and from which row to start the import, can be saved and reused. The import configuration is saved by clicking the **Save mapping as import module** button prior to performing the import. Next time the Import Tags dialog is opened, the configuration is available from the **Import module** drop-down list. Saving import configurations is valid only for files in text or Excel format.

4.13.3 Tag Import Example

If the controller in the project is changed to a system with other names for the tags used, these must be changed:

1. Click on **Tags** in the Project Explorer.
2. Click on the arrow to the right of the **Import/Export** button to display the available options of import and export operations.
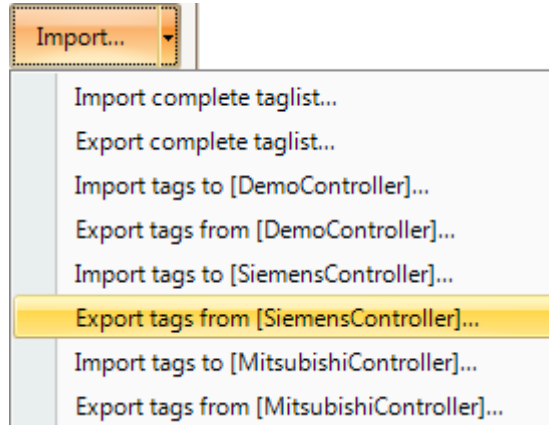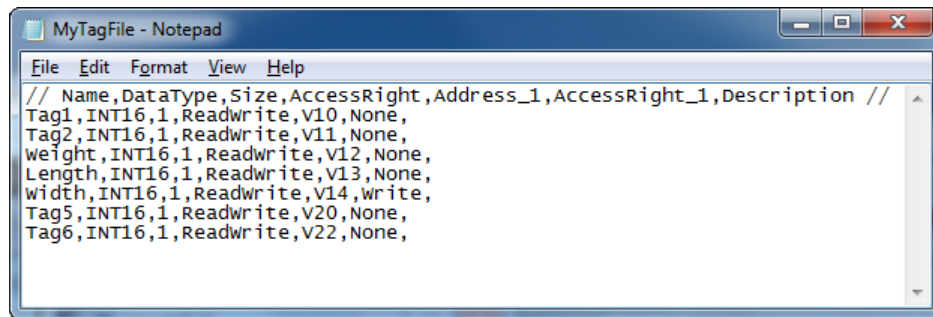
3. Select **Export tags** from the desired controller to export the list to a text file or an Excel sheet. Enter a file name and browse to a location. Define how the to separate the columns and which columns to export. Use the arrows if you want to change position of the columns.
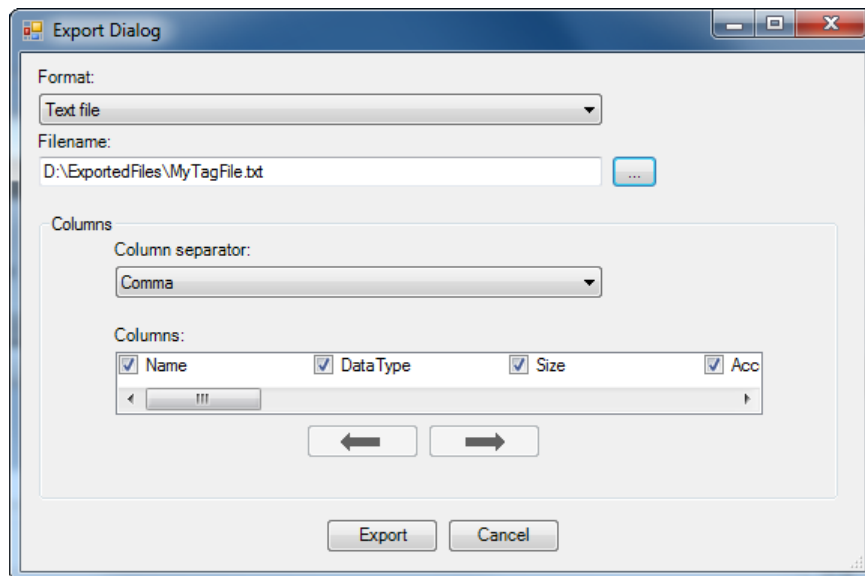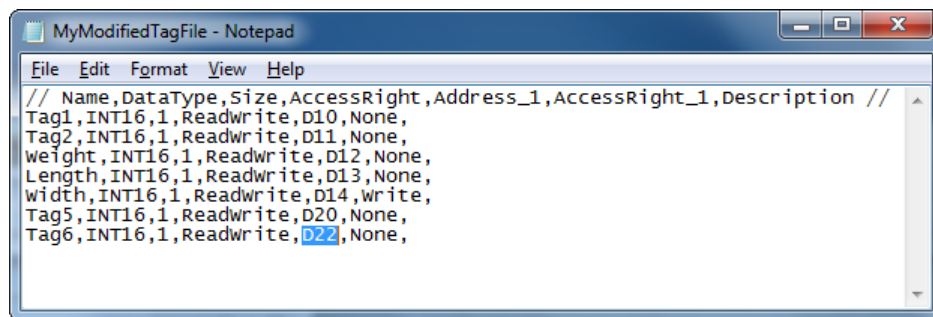
4. Click **Export**.

5. Open the text file in a text editor or a spread sheet application.

```
MyTagFile - Notepad
File  Edit  Format  View  Help
// Name,DataType,Size,AccessRight,Address_1,AccessRight_1,Description //
Tag1,INT16,1,ReadWrite,V10,None,
Tag2,INT16,1,ReadWrite,V11,None,
Weight,INT16,1,ReadWrite,V12,None,
Length,INT16,1,ReadWrite,V13,None,
Width,INT16,1,ReadWrite,V14,Write,
Tag5,INT16,1,ReadWrite,V20,None,
Tag6,INT16,1,ReadWrite,V22,None,
```

6. Change all current tags to corresponding tags used in the new controller, and save the file.

```
MyModifiedTagFile - Notepad
File  Edit  Format  View  Help
// Name,DataType,Size,AccessRight,Address_1,AccessRight_1,Description //
Tag1,INT16,1,ReadWrite,D10,None,
Tag2,INT16,1,ReadWrite,D11,None,
Weight,INT16,1,ReadWrite,D12,None,
Length,INT16,1,ReadWrite,D13,None,
Width,INT16,1,ReadWrite,D14,Write,
Tag5,INT16,1,ReadWrite,D20,None,
Tag6,INT16,1,ReadWrite,D22,None,
```

7. Select **Import tags** to the desired controller and browse to the updated file. If desired, select a row from which to start the import. It is possible to rebind columns before importing to the project, and to exclude columns. These settings can be saved by clicking the **Save mapping as import module** button, and will then be made available for selection from the Import module drop-down list at next import.
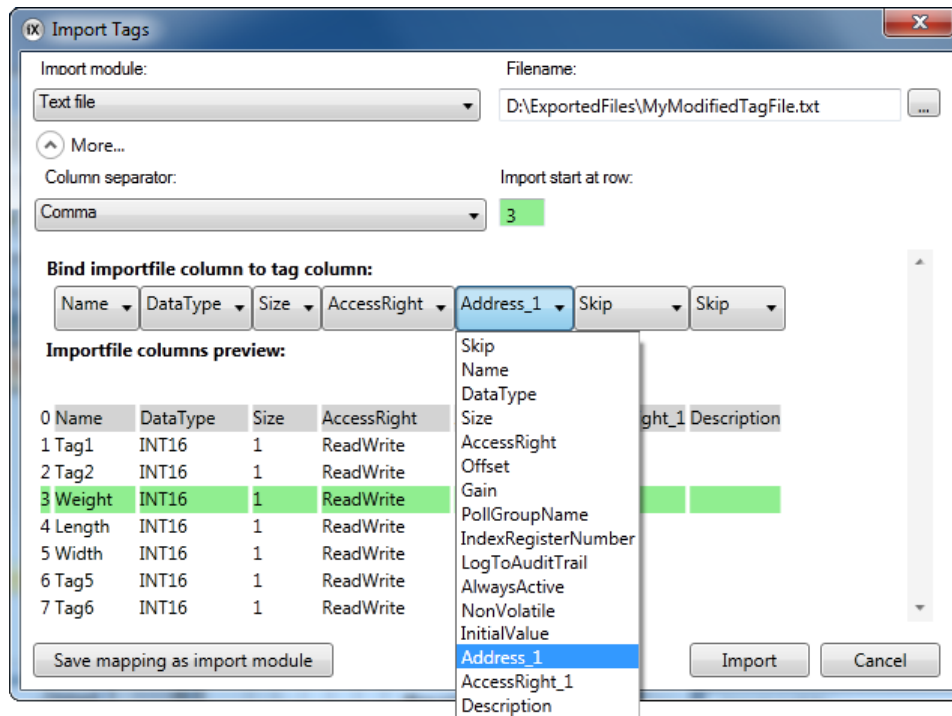
8. Click **Import**. It is now possible to deselect import of some tags.
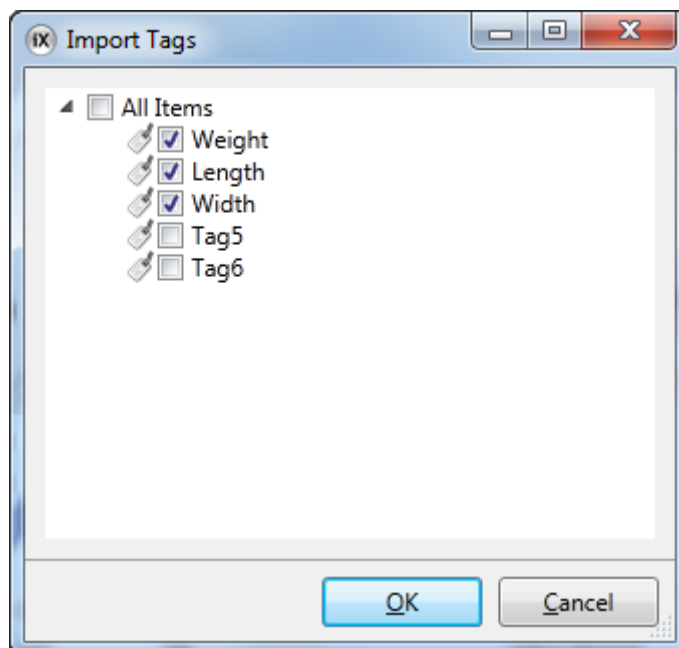
9. In case there are name conflicts, these can be handled by changing the name, overwriting, merging or skipping conflicting items. The actions taken are saved to a log file for future reference.
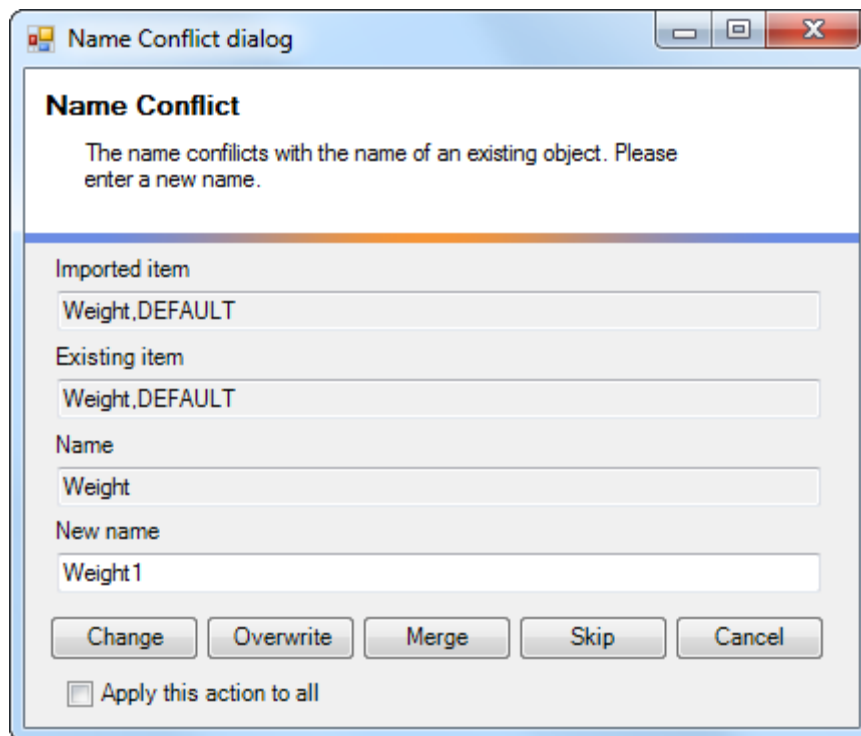
Note:
If you select **Overwrite**, all settings for the existing tag will be overwritten. If you select **Merge**, existing actions for the tag will be kept.

The imported tags are now added to the current project.

4.13.4 Importing Tags from the Command Line

Tags can also be added to the current iX Developer project using a command line switch, by entering *[Executable file] "Project location]\[Project name].neoproj"/ImportTags="[Tag file location]\Tag file"*. In this case, no application window is displayed, as long as the import is successful.

Example in Windows XP

To open the project named Test1 with the tag file VarFile.txt in the directory C:\MyProjects, run iX Developer from the command line with the following arguments:

NeoIDE.exe "C:\MyProjects\Test1\Test1.neoproj"/ImportTags="C:\MyProjects\VarFile.txt"

In Windows Vista, the complete path (within quotes) to the executable file has to be included.

Removing Existing Controller Tags at Import

The */ImportTags* command line switch may be used together with the switch */RemoveTagsNotInFile*. This will remove all existing tags in the controller that are not included in the import file.

4.14 Filtering Tags
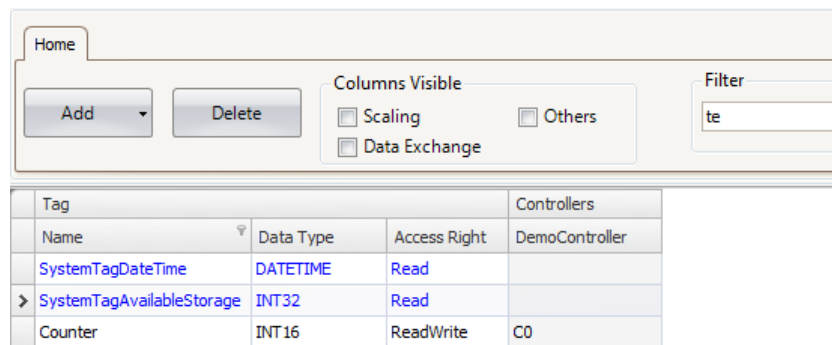
A selection of tags can be displayed using the Filter Builder. Several conditions can be used, and the different values/operators that can be used for making conditions are displayed when clicking on the condition building blocks.

Operands and operator nodes are context sensitive. Name strings are not case sensitive. Values in the filter builder are saved individually for each function during an edit session. A filter is built as a logical expression tree. A filter typically has operations on individual column values as leaf nodes and logical operators serve as main nodes.

Perform the following steps to create a filter:

1. Click on **Tags** in the Project Explorer.

    The tag list is displayed in the desktop area.

2. Select **Show Selection** from the Show Selection/Show All button to build a filter.

It is also possible to filter tags depending on tag name by typing in the **Filter** box. The filtering affects the tags list directly as letters are entered in the box.

Only tags including "te" in their names are displayed

The filter indication when having entered "te" in the Filter box

Related information
Filtering Items

4.15 Tag Format

The following tag formats (data types) are available in the dialog for a tag, provided that the selected driver or OPC server supports the tag format.

Format	Data type	Range
	DEFAULT	Treated as INT16
Bit	BIT	0 and 1
Bit	BOOL	Values can be "false" (0) or "true" (1)[1]
Signed 16-bit	INT16	-32,768 – +32,767
Unsigned 16-bit	UINT16	0 – +65,535
Signed 32-bit	INT32	-2,147,483,648 – +2,147,483,647[2]
Unsigned 32-bit	UINT32	0 – +4,294,967,295[2]
Float with exponent, 64-bit	DOUBLE	1.7E308[1]
Float with exponent, 32-bit	FLOAT	±3.4E38[1]
7 x Signed 16-bit	DATETIME	An Analog Numeric object can present the time format[1]. Syntax[3]: **<yyyy-MM-dd hh:mm:ss>** 7 consecutive registers are used to store a time: **Dx**, seconds, 0-59 **Dx+1**, minutes, 0-59 **Dx+2**, hours, 0-23 **Dx+3**, day, 1-31 **Dx+4**, month, 1-12 **Dx+5**, year, 100-9999 **Dx+6**, weekday, 1-7
String	STRING	Storing character strings in tags. For this data type, size can be selected.

1. Controller representation for this type is dependent on the driver software.
2. Regarding resolution of values in 32-bit format: both iX Developer and the operator panel can handle up to six decimal digits of an integer 32-bit value. Remaining digits will be truncated or changed into zeros.
3. **yyyy**=year, **MM**=month, **dd**=date, **hh**=hour, **mm**=minute, **ss**=second.

For use in script code, the following table shows the type representation in C#. This is used when there is a need to do a type casting, for example in calculations that are using overloaded operators.

C# type	Data type	Description
	DEFAULT	Treated as INT16
uint	BIT	C# does not have a "bit" type. In the controller this is a bit.
bool	BOOL	C# bool type that can have the values "false" (0) and "true" (1)
int	INT16	16 bit, -32,768 – +32,767
uint	UINT16	16 bit, 0 – +65,535
int	INT32	32 bit, -2,147,483,648 – +2,147,483,647
uint	UINT32	32 bit, 0 – +4,294,967,295
float	FLOAT	32 bit, ±3.4E38
double	DOUBLE	64 bit, ±1.7E308
DateTime	DATETIME	7 elements are used to store a time
string	STRING	Storing character strings in tags

Related information
Script

4.16 Aliases

An alias should be treated as a property of a screen and can act as a stand-in for a tag. The alias can be used to reference a tag wherever that tag can be referenced directly. When used, iX Developer will act the same as if the original tag had been used. Each screen using aliases can be configured with a different instance different tag references.

Using aliases will enable reuse of screens without having to create a new duplicated screen.

4.16.1   Creating Aliases

Aliases are handled by clicking **Aliases** on the desktop view mode tab.
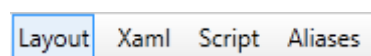
Desktop view mode tabs
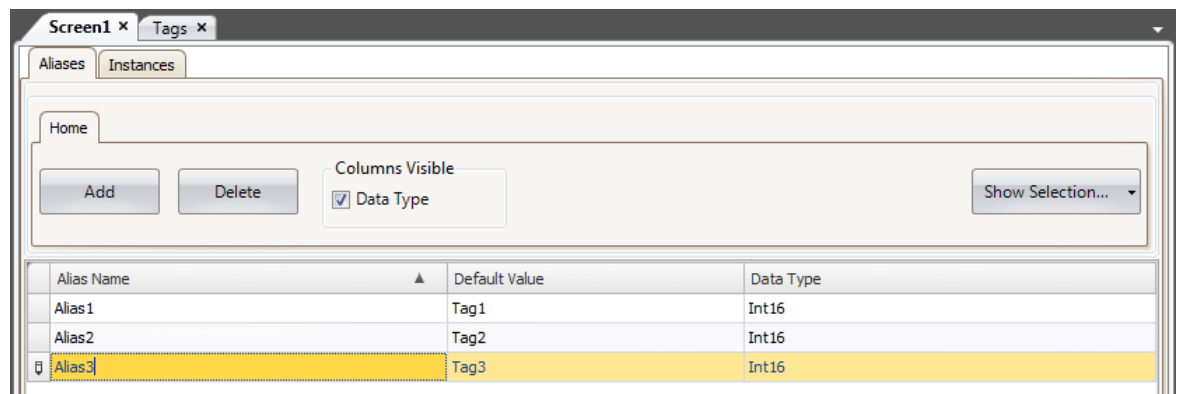
Clicking the **Add** button adds an alias to the list. Make settings for each added alias. When predefined values are available, these can be selected by clicking the arrow at the right of the table cell.

Basic Settings

Enter **Alias name** and **Default value** for the alias. If no default value is entered, the alias will be treated as a local screen variable.

Parameter	Description
Alias Name	Symbolic name. The name can be any alphanumeric string, beginning with a letter (a-z, A-Z).
Default Value	Selection of tags to create aliases of. If no tag is selected the alias will be treated as a local screen variable.
	The **default value** will be the default value for the alias in the created instances. It will also be the reference for the default instance. See *Instances*.
Data Type	Selection of data type. This setting is enabled by first clicking the Data Type checkbox.
	The data type will by default reflect the data type of the selected Default Value.

4.16.2   Instances

The aliases created in the project are presented in a table on the instances tab. It is possible to change the value for the aliases in this tab, in order to override the default value configured in the aliases tab.

If a value for an alias is changed in the instances tab, the new value is displayed with bold text to indicate that it differs from the alias' **default value**.

To revert back to the **default value**, delete the value followed by pressing [**Enter**].

The **default value** can be changed in the aliases tab.
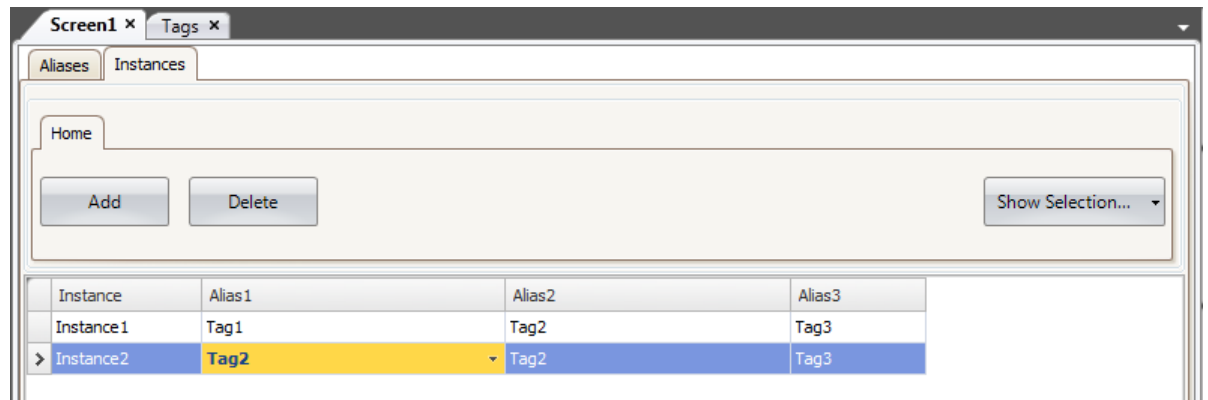
Parameter	Description
Instance	Symbolic name. The name can be any alphanumeric string, beginning with a letter (a-z, A-Z).
Aliases	The names of all aliases created in the project will be displayed in the columns headers. The values connected to the aliases will be listed in each columns.

Using the "Show Screen" Action with Instances

The "**Show Screen**" action can be used to display a specific screen. For screens that have aliases, it is also possible to select which instance of the screen to use, and thereby also which set of tags to assign to the screen's aliases.

The **Default** option is a virtual instance configured with the alias' default values.
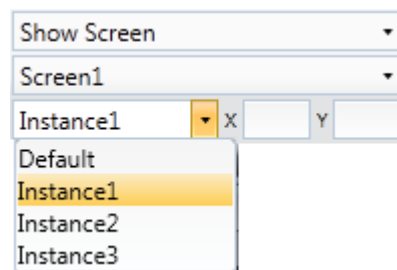
"Show Screen" Action Parameters.

Related information
Action Groups

Limitations

Aliases do not support array tags.

Aliases can not be used in connection with expressions.

Aliases can not be used with the Trend Viewer object.

No scripting support for valuechanged event on an alias.

5   Controller

Operator panels can be connected to many types of automation equipment, such as PLCs, servos, and drives.

The expression *controller* is used as a general term for the connected equipment.

It is possible to connect a tag to more than one controller simultaneously.
This makes it possible to use the same application with different controllers.
Controllers can be enabled and disabled in runtime. A project can contain multiple controller connections.

A project can also be connected to an external OPC server.

Related information
Multiple Controllers
External OPC Server
Internal Tags

5.1 Adding a Controller

When a new project is created, one of the steps is to choose which controller brand and model to connect to. Clicking on **Tags** in the Project Explorer, selecting the **Controller** tab displays the Controller configuration page.
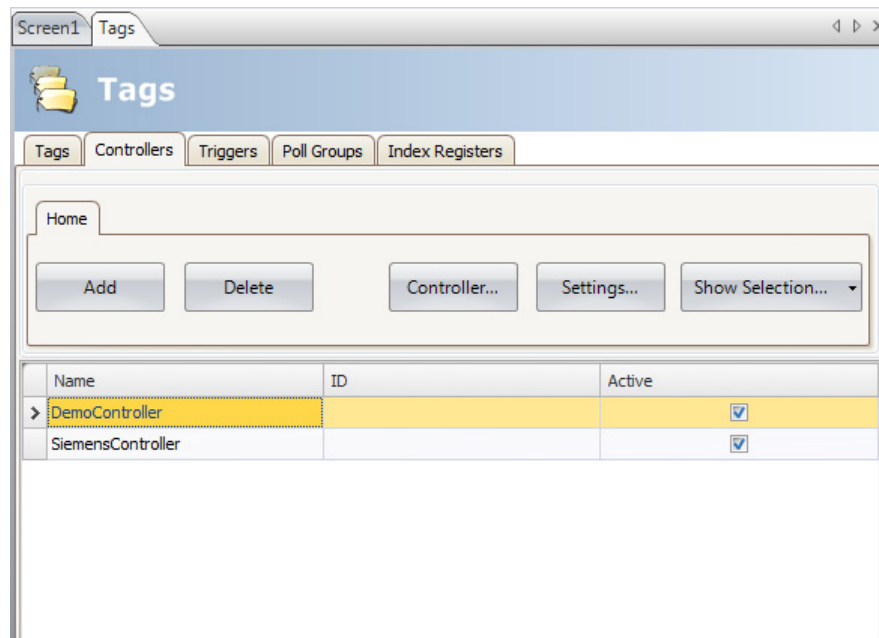
Parameter	Description
Name	Symbolic name. The name can be any alphanumeric string, beginning with a letter (a-z, A-Z).
ID	Optional short name for controller; displayed in functions such as Data exchange.
Active	Selection of which controllers are to be active in runtime. Change of active controllers can also be performed in runtime using an action or script.

Controllers are added and deleted using the **Add** and **Delete** buttons.

Clicking **Controller** allows selecting another brand for the selected controller, and **Settings** displays settings for the selected driver.

Clicking **Help** in the driver settings dialog opens the help file for the driver.

Related information
Select Controller

5.1.1   Notify Window

If a communication error between the application and the controller is detected in runtime, this will result in a communication error, and a Notify Window will be displayed in the operator panel. The Notify Window is a floating window that starts in the upper left corner of the screen. It is also used to display the alarm indicator.

The Notify Window can be moved to any position on the current screen.

5.2 DEMO Controller

The DEMO controller is used and selected in the same way as a regular controller driver, to design and test a project directly on the development PC without connection to an external controller.

A number of predefined tags can be selected:

Data type	Tag ID	Description
BIT, BOOL	M0 - M99	Digital tags
INT16, UINT16	D0 - D99	Analog tags

The DEMO controller also includes counters and digital tags with predefined functions:

Data type	Tag ID	Description
BIT	M100-M104	Toggles OFF and ON every 1,000 ms
INT16	C0-C4	Counts from 0 to 100 with 1,000 ms interval by default; min / max values can be defined

5.3 External OPC Server

OPC is an industrial standard created in a collaboration of automation hardware and software suppliers. The standard defines methods for exchanging real-time automation data between PC-based clients using Microsoft operating systems.

An OPC server acts as a controller towards iX Developer.

5.3.1 OPC Classic Client

Settings for the OPC Classic client is made in the **Choose Controller** dialog.

Select **Localhost** when accessing an OPC server running in your local PC.

Select **Remote Server** when accessing an OPC server in the network environment, and type the IP address of the OPC server, or click **Browse** to select the OPC server. Clicking **Refresh** makes it possible to select the OPC server from the OPC Server drop-down list next time.

Add OPC server tags by clicking the small arrow at the right-hand side of the **Add** button on the Tags tab:
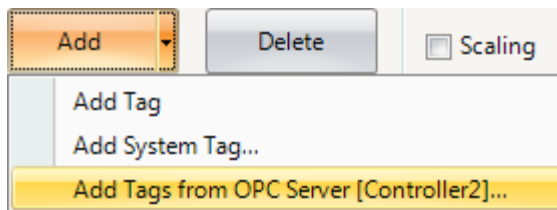
To validate tags from a remote OPC server in the iX Developer project, return to the Controllers tab and select the remote OPC server once more, and click **OK**.

This procedure is necessary as the tags in the remote OPC server may have been changed since they were selected on the Tags tab.

5.3.2 OPC UA Client

Settings for the OPC UA client can be made in the **Choose Controller** dialog. Open the dialog by clicking the **Add** button on the Controllers tab.

Enter the URL to the **OPC UA** server.

Controller Settings Dialog

The settings dialog for the OPC UA Server is available by clicking the **Settings** button on the Controllers tab.

Parameter	Description
URL	Enter URL to the UPC UA server. Only UA TCP Binary is supported which means the URL must start with "opc.tcp://"
Authentication settings	Choose whether to use anonymous login or enter a user name and password to enable a secured login.
Namespace settings	This setting offers a possibility to declare a namespace URI and bind it to a prefix.
Tag View settings	**Namespace separator character**: Enter a character to be used to separate the namespace URI prefix from the tag ID when presented on the tag tab.
	Default namespace: Select the default namespace URI prefix to be used when entering a tag on the tag tab.

Add Tags from OPC UA Server

Add OPC UA server tags by clicking the **Add** button on the Tags tab.

Enter the Tag ID along with the preferred namespace URI prefix, defined in the controller settings dialog. Make sure to separate the values with the separator character.

Example: "**NS3:tag**" .

The Tag ID, in the example above, corresponds with **BrowseName** in the OPC UA server.

If no namespace URI prefix is entered, the default namespace URI prefix as defined in the controller settings dialog will be used.

No prefix is needed when connecting to an iX server.
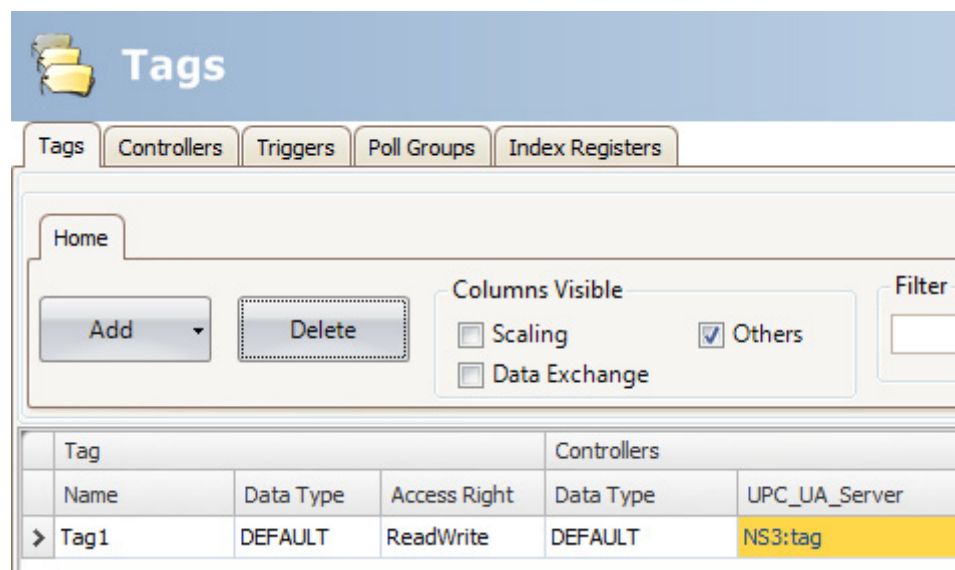
Note:
Array tags are not supported by OPC UA controllers.

Supported Functions

Data Access (UA part 8) is supported.

All variables of data types that are used in iX Developer is supported. Read / write of objects, structs and arrays is not supported.

Limitations

Limitation	Description
Browsing	Browsing of servers / nodes is not supported.
Address input in GUI	**NameSpace** and **BrowseName** is used to identify a variable in the server.
	If this combination is not unique, the client cannot identify the variable.
	This is not a problem when connecting to an iX server, since all tag names (variables) are unique.

Related information
OPC UA Server

5.4 Updating Drivers

Drivers can be updated via Internet or from file.

Each driver takes up approximately 500 kB. To use the downloaded drivers, iX Developer must be restarted.

5.4.1 Updating Drivers from Internet

A web browser is not required to update drivers via an Internet connection. A list of drivers that can be downloaded via Internet is displayed when the connection is established.

1. Click on the iX Developer button and select **Update Drivers /From Internet** to update available drivers to the latest version, or to install new drivers.
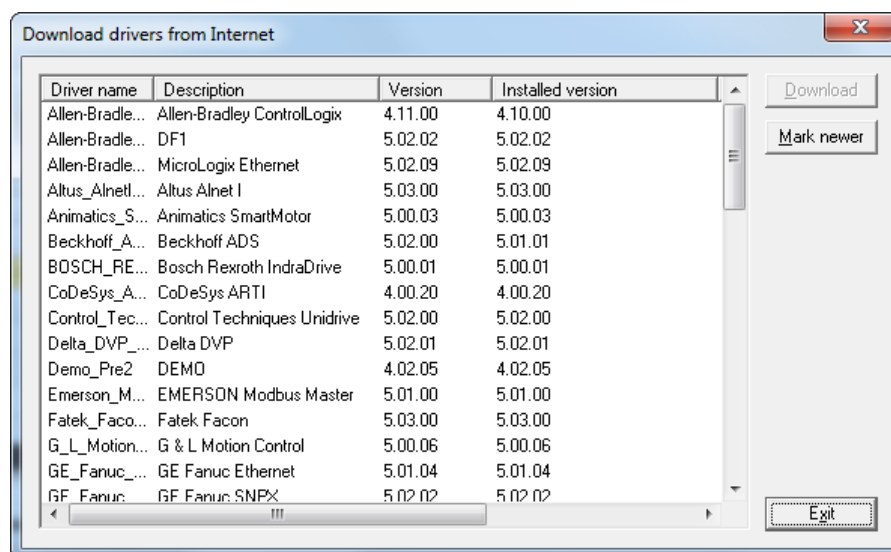
The list shows the version number for the available drivers and the version number for the installed drivers.

2. Select the driver(s) to install in iX Developer.

    The command **Mark Newer** selects all drivers available in newer versions than the ones installed, including all drivers that are not installed.

3. Click **Download**.

Related information
File Menu

5.4.2   Updating Drivers from File

When updating drivers from file only one driver at the time is updated. This can for example be used to revert to an older driver.
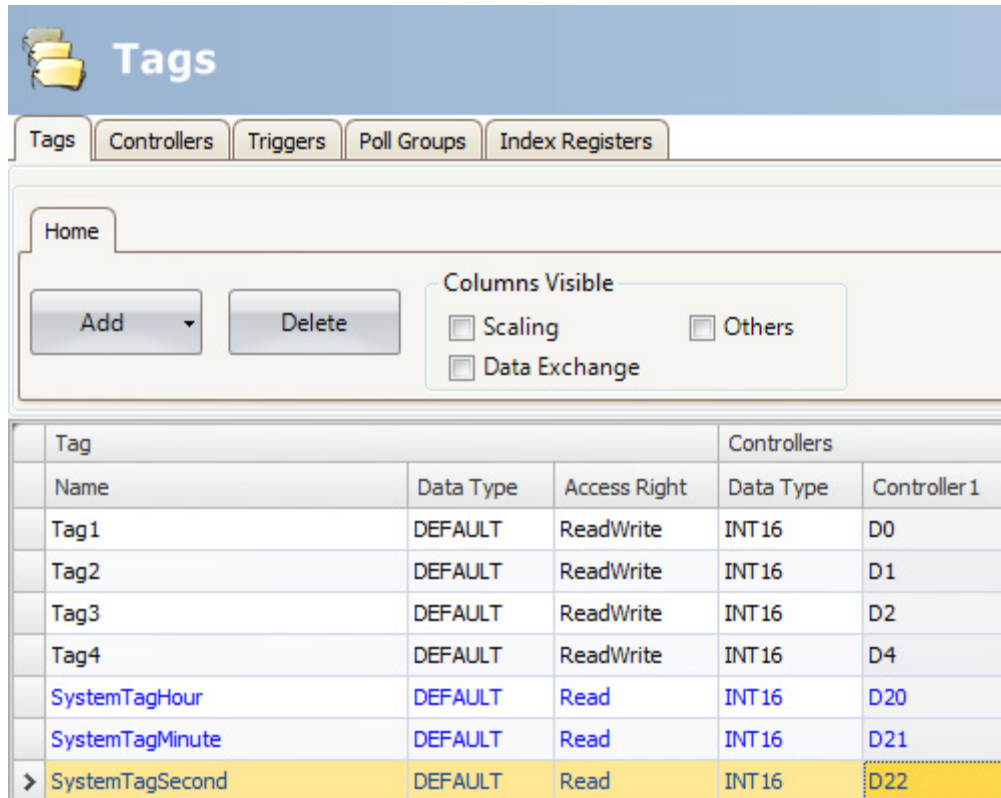
To install a new driver from file:

1. Click on the iX Developer button and select **Update Drivers /From File** to update available drivers to the latest version.
2. Browse to the folder where the driver files are located.

   A list of all drivers that can be installed is now shown.
3. Select the driver to install in iX Developer.

   Mark Newer selects the driver when there is a newer version than the currently installed.
4. Click **Install**.

Related information
File Menu

5.5   Synchronizing the Controller Clock

In order to synchronize the controller clock with the iX Developer project, the system tags hour, minute, second, need to be added to the project from the Tags configuration page. These are then connected to addresses in the controller.

When the value is changed in any of the system tags, they will be written to the controller.

Related information
System Tags

6   Web Server

The Web Server adds the possibility to use a server-side Application Programming Interface (API) to interact with a running project. The API operates over the HyperText Transfer Protocol (HTTP) and can be accessed from any compatible device. In addition, the web server can host web pages and related media to be consumed by web browsers.

The bundled Javascript Software Development Kit (SDK) makes it easy to build interactive web pages that utilize the API.

Login settings for the web server are available from Server group on the System ribbon tab.

6.1   Web Server Configuration

6.1.1   Web Site Configuration

For the web server to be able to host web pages, the content of the web site must be transferred using FTP to a folder named "WebSite" residing in the FTP root directory.

The FTP server must be enabled and the web content can only be transferred after the application has been deployed and started on the panel.

6.1.2   Login Form

The web server provides a default login form that is used for clients to authenticate. It is also possible to override the default and provide a custom web page that matches your overall design.

To override the login form, create an HTML file called "login.html" and place it in a folder named "Website" in the root of the FTP root directory.

This file will be used instead of the default when clients access the address "**/login**".

The Javascript SDK provides tools for creating the form using a few conventions.

Note that this only applies if you have authentication enabled in the web server settings.

Authentication Script

Reference the authentication script

```
<script src='/assets/authentication.js' type='text/javascript'></script>
```

Username and Password Field

Mark the username and password field with the predefined id's "**ix-username**" and "**ix-password**":

```
<input id="ix-username" type="text" maxlength="20" required="true" />
<input id="ix-password" type="password" maxlength="20" required="true" />
```

Login Button

Mark a login button with the predefined id "**ix-login**". Any element can be used.

```
<button id="ix-login">LOGIN</button>
```

Invalid Credentials

Optionally add an element with the predefined id "**ix-invalid-credentials**" that will be shown if the login fails.

```
<span id="ix-invalid-credentials">Failed to login, try again.</span>
```

6.2   Javascript SDK

6.2.1   Javascript SDK Overview

The JavaScript SDK provides a rich set of client-side functionality for accessing the server-side API calls. This enables the possibility to build highly interactive web pages that uses tags defined in the project.

1. The first step to build an interactive web page is to reference the SDK by inserting the following script into to the header of the html.

```html
<script src='/assets/iX.js' type='text/javascript'></script>
```

Note:
The SDK is bundled with jQuery 1.7 (*http://jquery.com/*).

2. Next, the SDK needs to be initialized. In its simples form the following lines need to be added after the reference to the SDK:

```html
<script type="text/javascript">
        iX.setup({});
</script>
```

This will make it possible to use the defined tag metadata attributes to incorporate the project tags into the web page.

The following line will, for example, insert the value of Tag1 into the html document:

```html
<span data-ix-tag="Tag1"></span>
```

Tag Value or Properties

To insert the current value of a tag or tag property into a html element, the **data-ix-tag** attribute can be used. The value of the attribute should be the name of the tag as defined in the project and the property of a tag. Separate the names with a dot ".".

If no property is provided, the value of the tag is used.

```html
1   <span data-ix-tag="Tag1">Replaced with Tag1 value</span>
2   <span data-ix-tag="Tag1.dataType">Replaced with the datatype of Tag1</span>
3
```

Formatting

Formatting of a tag value can be applied to a tag of numeric type or a datetime. The format is defined using the **data-ix-format** attribute.

```
1   <span data-ix-tag="SystemTagDateTime" data-ix-format="Y-M-d" />
2   <span data-ix-tag="NumericTag" data-ix-format="0.000" />
3
```

The following table describes the custom date and time format specifiers and displays a result string produced by each format specifier.

Tag value	Description	Result
d	The day of the month, from 01 through 31.	Tuesday, February 07, 2012 12:27:24 PM-> 07
D	The abbreviated name of the day of the week (in english)	Tuesday, February 07, 2012 12:27:24 PM-> Tue
j	The day of the month, from 1 through 31	Tuesday, February 07, 2012 12:27:24 PM-> 7
l	The full name of the day of the week	Tuesday, February 07, 2012 12:27:24 PM-> Tuesday
S	Ordinal number for the date of the month (eg -st, -nd, -rd, -th etc)	Tuesday, February 07, 2012 12:27:24 PM-> th
w	Day of the week as a number.	Tuesday, February 07, 2012 12:27:24 PM-> 2
z	Day of the year	Tuesday, February 07, 2012 12:27:24 PM-> 37
W	Week of the year	Tuesday, February 07, 2012 12:27:24 PM-> 05
F	The full name of the month.	Tuesday, February 07, 2012 12:27:24 PM-> February
m	Gets the day of the month from 00 to 31	Tuesday, February 07, 2012 12:27:24 PM-> 02
M	The abbreviated name of the month.	Tuesday, February 07, 2012 12:27:24 PM-> Feb
n	The month of the year 1 to 12	Tuesday, February 07, 2012 12:27:24 PM-> 2
t	Number of days in the month	Tuesday, February 07, 2012 12:27:24 PM-> 29
L	Is leap year, 1 if true, otherwise 0	Tuesday, February 07, 2012 12:27:24 PM-> 1
Y	The year as a four-digit number.	Tuesday, February 07, 2012 12:27:24 PM-> 2012

Tag value	Description	Result
y	The year, from 00 to 99	Tuesday, February 07, 2012 12:27:24 PM-> 12
a	The AM/PM designator lower case.	Tuesday, February 07, 2012 12:27:24 PM-> pm
A	The AM/PM designator upper case.	Tuesday, February 07, 2012 12:27:24 PM-> PM
g	The hour, using a 12-hour clock from 1 to 12.	Tuesday, February 07, 2012 12:27:24 PM-> 12
G	The hour, using a 24-hour clock from 0 to 23.	Tuesday, February 07, 2012 12:27:24 PM-> 12
h	The hour, using a 12-hour clock from 01 to 12.	Tuesday, February 07, 2012 12:27:24 PM-> 12
H	The hour, using a 24-hour clock from 00 to 23.	Tuesday, February 07, 2012 12:27:24 PM-> 12
i	The minute, from 00 through 59	Tuesday, February 07, 2012 12:27:24 PM-> 27
s	The second, from 00 through 59.	Tuesday, February 07, 2012 12:27:24 PM-> 24
X	The milliseconds, from 000 through 999.	Tuesday, February 07, 2012 12:27:24 PM-> 000
O	Hours offset from GMT, with a leading zero for a single-digit value.	Tuesday, February 07, 2012 12:27:24 PM-> +0100
Z	Timezone offset, in seconds	Tuesday, February 07, 2012 12:27:24 PM-> 3600
q	Quarter	Tuesday, February 07, 2012 12:27:24 PM-> 1

The following table describes the custom numeric format specifiers and displays sample output produced by each format specifier:

Tag Value	Description	Result
0	Digit	Value: 4, Format: 00 -> 04 Value: 4.5, Format: 0.00 -> 4.50 Value: .1, Format: 0.00e00 -> 1.00e-01
#	Digit, zero shows as absent	Value: 4, Format: ## -> 4
.	Decimal separator or monetary decimal separator	Value: 4.5, Format: 0.00 -> 4.50
E	Separates mantissa and exponent in scientific notation	Value: 5.55, Format: 0.0E+00 -> 5.6E+00 Value: 1000000, Format: 0.0E+00 -> 1.0E+06
,	Grouping separator	Value: 1000, Format: 0,0 -> 1,000 Value: 1000, Format: 0,# -> 1,000
;	Separates positive and negative subpatterns	Value: -5, Format: plus;minus -> minus Value: 5, Format: plus;minus -> plus Value: 0, Format: plus;minus;zero -> zero Value: -5, Format: 0;(0.0);0.000 -> (5.0)
[]	Custom subpatterns	Value: 700, Format: [>500]###.##;[<1000]###;###.# -> 700.00 Value: 400, Format: [>500]medium;[>1000]###;###.# -> 400 Value: 1001, Format: [>2000]high;[>1000] normal;[>500]low -> normal

Setting Tag Value

To set the value of a tag, two elements must be defined.

The first element have the **data-ix-setter** attribute on the position where the tag value can be entered.

The second element has the **data-ix-submitbutton** attribute used to submit the new tag value.

The **data-ix-submitbutton** attribute takes the tag names, separated by comma ",", to be submitted when the button is clicked. To submit all tags an asterisk "*" can be used.

```
1   <input type="number" data-ix-setter="Tag1" />
2   <input type="button" data-ix-submitbutton="Tag1" />
3
```

iX-invalid Style

Basic validation is done on input values. If the value cannot be assigned to the Tag then the value will not be set. To indicate an invalid value the framework will set the CSS class "ix-invalid" on the element. This can then be used to provide the user with visual cues of the invalid state. For example, the following CSS snippet adds a red border around the input when the input is incorrect:

```
<style type="text/css">
    .ix-invalid {
        border: 1px solid red;
    }
</style>
```

Refresh Modes

To control how tag values are refreshed, the **data-ix-refresh** attribute can be used with elements defined with **data-ix-tag** or **data-ix-setter**.

It is recommended to use the least frequent interval that is required by your project.

```
1   <span data-ix-tag="Tag1" data-ix-refresh="onetime" />
2   <span data-ix-setter="Tag2" data-ix-refresh="none" />
3
```

Advanced

iX.setup

After loading the JavaScript SDK, call iX.setup to initialize the framework. This will wire the tag metadata attributes that you have defined in html and will start periodical polling if this is needed.

Parameters:

Name	Type	Description
options	Object	See the "Options" table below.

Options:

Property	Type	Description	Argu-ment	De-fault
refreshInterval	Int	The refresh interval of tag values in ms. For performance reasons it is recommended not to use low values for this setting.	Optional	5000
refreshMode	String	Overrides the default refresh mode of tag values. Accepted modes are: **none**, **onetime**, **interval**.	Optional	**one-time**
refreshModeSetter	String	Overrides the default refresh mode of tag setters. Accepted modes are: **none**, **onetime**, **interval**.	Optional	**one-time**

Example:

This example will set periodical refresh interval to 10 seconds, set the refresh mode of html elements marked with the **data-ix-tag** attribute to interval and set the refresh mode of html elements marked with the **data-ix-setter** attribute to refresh only once.

```
iX.setup({ refreshInterval: 10000, refreshMode: 'interval',  refreshModeSetter: 'onetime' });
```

iX.refreshElements

Refreshes the tag values of the provided elements manually. The attribute data-ix-refreshbutton can be used to achieve the same function.

Note:
Only an element that have been marked with a valid data-ix-tag or data-ix-setter can be refreshed. The value will be refreshed regardless of the refresh mode set on the element.

Parameters:

Name	Type	Description
elementIds	Array	A list of strings containing the html element ids. If not provided all tag elements are updated.

Example:

This example show how you programmatically can achieve the same behavior as marking an element with **data-ix-refreshbutton**. In the example, the html element with id "tag1Id" will be updated when the element with id "refreshButton" is clicked. This example uses jQuery (bundled with iX.js) to bind to the click event of the "refreshButton"

In html:

```html
<button id="refreshButton">Refresh</button>
<span id="tag1Id" data-ix-tag="Tag1"></span>
```

In Javascript:

```javascript
iX.setup({});

$(document).ready(function() {
    $('#refreshButton').click(function() { iX.refreshElements(['tag1Id']); });
});
```

iX.refreshTags

Refreshes the tag values of the provided tag names manually.

Note:
Only tags used in conjunction with data-ix-tag or data-ix-setter or created by iX.createTag can be refreshed.

Parameters:

Name	Type	Description
tagName	String	(Optional) A list of strings containing the names of the tags to refresh. If not provided all defined tags are refreshed.

Example:

This example show how to programmatically refresh the tag values of "tag1" and "tag2". When refreshButton is clicked the "tag1" and "tag2" will be refreshed even if the refresh mode was set to 'none' when creating the tag.

```
iX.setup({});
iX.createTag('tag1', function() { alert('tag1 updated') }, 'none');
iX.createTag('tag2', function() { alert('tag2 updated') }, 'none');

...

$('#refreshButton').click(function() { iX.refreshTags(['tag1', 'tag2']) });
```

iX.createTag

Creates a representation of a server-side tag on the client-side.

Parameters:

Name	Type	Description
tagName	String	The name of the tag in the project to create a connection to.
valueRefreshedCall-back	Function	Optional: Function that is called when the tag value has changed. The function is not called when the tag value is the same as last checked. Function may take two optional parameters. The first parameter is the tag that changed the value, the second the old values of the tag before it changed.
refreshMode	String	Optional: Overrides the default refresh mode of tag. Accepted modes are: **none**, **onetime**, **interval**. Note that the more frequent refreshMode is always used. For example, if the tag is created with **onetime** but an element defines **data-ix-refreshmode="interval"** for the same tag then the tag will have a refreshMode of **interval**.

Returns:

Type	Description
iX.Tag	An instance of iX.Tag. See *Tag Class* for a complete description.

Example:

This example show how to display a warning if the value of "tag1" exceeds 70. The example is using jQuery (bundled with iX.js) to hide and show the warning box.

```
iX.setup({});

function onTag1Updated(tag, oldProperties) {
    if(tag.value > 70)
        $('#warningBox').show();
    else
        $('#warningBox').hide();
};

iX.createTag('tag1',  onTag1Updated, 'interval');
```

iX.getTag

Retrieves a tag by name.

Parameters:

Name	Type	Description
tagName	String	The name of the tag to retrieve.

Returns:

Type	Description
iX.Tag	An instance of **iX.Tag**. See *Tag Class* for a complete description.

Example:

This example shows how to retrieve a instance of a tag after it has been created. "tag1" and "tag1Again" will refer to the same tag instance.

```
iX.setup({});
var tag1 = iX.createTag('tag1', function() {}, 'interval');
var tag1Again = iX.getTag('tag1');
```

iX.commitTagValue

Commits any unsaved tag values that was changed using **iX.Tag.setValue** for the provided tag names.

Parameters:

Name	Type	Description
tagNames	Array	(Optional) A list of strings containing the names of the tags to commit. If not provided all defined tags are committed.

Example:

This example show how to commit changes to tag values on the server in a batch. Note that the value must match the data type of the tag on the server.

```
iX.setup({});
var tag1 = iX.createTag('tag1', function() {}, 'interval');
var tag2 = iX.createTag('tag2', function() {}, 'interval');

...

tag1.setValue(1234);
tag2.setValue(4567);
iX.commitTagValues(['tag1', 'tag2']);
```

Tag Class

iX.Tag is a class representing a Tag. The methods and properties of a Tag must be accessed through an instance of **iX.Tag** and not the class itself. Typically a Tag instance obtained using **iX.createTag** or **iX.getTag** methods.

iX.Tag Properties

Name	Type	Description
isDirty	Bool	Flag indicating if the local value has been changed.
tagProperties	Object	An object with data retrieved from the server. See tag metadata below. Note that these are only available after the first refresh. Also, only value is refreshed not all of the metadata.
value	Flexible	The value of the tag. The type of the value depends on the data type in the project. Never use this property to set the value.

Tag Properties:

Name	Type	Description
controllers	Array	An array of controller names that the tag is connected to.
description	String	The tag description, null if not defined.
dataType	String	The tag **dataType**, e.g **int16**, **datetime**, **default**
initialValue	Flexible	The initial value of the tag, null if not defined.
readonly	Bool	True if the tag is read only, otherwise false
quality	String	Indicates the quality of the tag value: **Unknown**, **Good**, **Bad**

iX.Tag.setValue

Set the tag value manually. To commit the value to the server side call **iX.commit**. It's vital that the value that is set corresponds to the data type of the tag.

Parameter:

Name	Type	Description
newValue	Flexible	The new value. Must match the data type of the tag.

iX.Tag.subscribeToChange

Adds a callback that is called when the server-side value has changed.

Parameter:

Name	Type	Description
fn	Function	The callback to register

iX.Tag.unsubscribeToChange

Unsubscribe from changes.

Name	Type	Description
fn	Function	The same callback that was used with **iX.Tag.subscribeToChange**

iX.Tag.refresh

Refreshes this tag. It is recommended to refresh tags in batch (using **iX.refresh**) instead of refreshing individual tags.

iX.Tag.commit

Commit this tags value if the value has been changed. It is recommended to commit tags in batch (using **iX.commit**) instead of refreshing individual tags.

6.3 Web Service API

6.3.1 RESTful Web Service API

iX Developer provides an API for interacting with tags. The tags can be read or written to using the HTTP verbs **GET**, **PUT** and **POST**. **GET** to read tags, **PUT** to set the value of a tag or **POST** to perform batch operation.

6.3.2 General

Accepted Types

It is possible to negotiate which media type that will be returned as a representation of the resource being retrieved. Use the Accept header in the request to specify the desired media type. text/html and application/json media types are understood as response formats for tag related resources.

A client request providing an unsupported media type will be responded with the following message: "406 - Not understood".

6.3.3 REST API

Tags Resource

Lists tags available in the project.

Verbs:	GET
Uri:	/Tags

Tags Resource Example

```
1   Request: GET /Tags
2   Response:
3   200 OK
4   ["Tag1","Tag2","Tag3","SystemTagDateTime","Tag4"]
5
```

Tag Resource

Resource for interacting with individual tags. To perform multiple operations on tags it is recommended to use **TagBatch** resource instead.

Verbs:	GET, PUT
Uri:	/tags/:name
GET params:	**Name:** The name of the tag, e.g, Tag1 SystemTagDateTime
PUT params:	**Name:** The name of the tag, e.g, Tag1 SystemTagDateTime
	Value: The value to set the tag to, e.g, 12, "foo"

Tag Resource Example

```
1   Request: GET /tags
2   Response:
3   200 OK
4   ["Tag1","Tag2","Tag3","SystemTagDateTime","Tag4"]
5
```

```
1    Request: PUT /tags/Tag1
2    Content-type: application/json
3
4    {
5        name: "Tag1",
6        value: 1234
7    }
8
9    Response: 200 OK
10
```

TagBatch Resource

The **TagBatch** resource is a resource used to perform batch operations on tags.

Verbs:	POST
Uri:	/tagbatch
POST params:	**includeMetadata**: boolean indicating if only the tag value or the full tag metadata should be fetched
	getTags: List of tag names to return
	setTags: List name, value pairs representing new tag values.

TagBatch Resource Example

```
1   Request: PUT /tagbatch
2   Content-type: application/json
3
4   {
5       "includeTagMetadata":true
6       "getTags": ["Tag1","Tag2"],
7       "setTags": [
8                     {
9                          "name":"Tag3",
10                         "value":45
11                    },
12                    {
13                         "name":"Tag4",
14                         "value":"foo"
15                    }
16                ]
17  }
18
19  Response: 200 OK
20  {
21      "tags":
22             [{  "quality":"good",
23                 "description":null,
24                 "name":"Tag1",
25                 "value":69,
26                 "datatype":"int16",
27                 "controllers":["Controller1"],
28                 "initialValue":null,
29                 "isOffline":false,
30                 "readonly":false
31             },
32             {
33                 "quality":"good",
34                 "description":null,
35                 "name":"Tag2",
36                 "value":45.56456,
37                 "datatype":"double",
38                 "controllers":[],
39                 "initialValue":45.56456,
40                 "isOffline":false,
41                 "readonly":false
42             }]
43  }
44
```

Note:
The returned value (**initialValue**) for tags of datetime type is presented in the ISO8601 format.
Example: "2008-09-22T14:01:54.9571247Z".
Dates are always returned in UTC, Coordinated Universal Time.

7   Objects

Static and dynamic objects that can be included in screens are available for selection from the Objects control group of the Home ribbon tab.
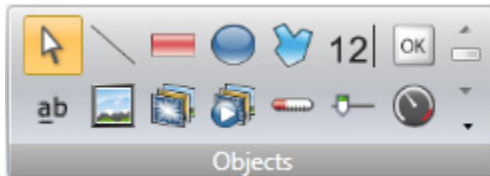
Not all objects are visible in the normal view, but are made available by scrolling. Clicking the expand button displays all the objects, categorized into groups.

An object can be created by clicking on it in the Objects group and then clicking on the screen, or by dragging it from the Objects group and dropping it on the screen. This creates the object with default size and appearance.

Dragging the object handles resizes it. Object appearance can be modified using the commands in the Format and Font control groups, but also by the controls on the Dynamics ribbon tab.

Settings for a selected object can be managed with the control groups on the General ribbon tab, and actions can be configured from the Actions ribbon tab.

It is possible to select a style to be used as the default style next time the same kind of object is added.

Related information
Default Appearance

7.1 Blinking Objects

Some objects can be displayed as blinking objects. Blinking is enabled for each object individually using the Property Grid.

Parameter	Description
BlinkInterval	Time in ms from when the object gets invisible one time until it gets invisible the next time. For operator panel projects, the minimum value is 400 ms, and it is recommended to use a blink interval of a multiple of 400 ms to ensure a harmonic blink experience (400 ms, 800 ms, 1200 ms, 1600 ms etc.). For PC targets, the minimum value is 200 ms. The default value is 1200 ms for all targets, and the maximum value is 10,000 ms for all targets.
IsBlinkEnabled	Checking the box makes the object blink

Security has higher priority than blink, and an object hidden by security cannot be made visible through blink.

The properties related to the blink function can be bound to tags using the Blink control on the Dynamics ribbon tab. This makes it possible to enable and disable blinking, and to change blink frequency based on a tag value.

Related information
General Group

7.1.1   Limitations for Operator Panel Targets

Blink is not supported for the following objects in operator panel targets:

Alarm Distributor Viewer	Check Box
Alarm Viewer	Combo Box
Audit Trail Viewer	Group Box
Database Viewer	List Box
Media Player	Progress Bar
Pdf Viewer	Radio Button
Web Browser	

Note:
Combining blink and visibility properties (e.g. using Dynamics) is not supported. This combination may lead to unexpected behavior.

Related information
General Group

7.2 Shapes

Static shapes such as Line, Rectangle and Poly Line can be used to draw background graphics. Tags can be linked to shapes under Dynamics, to change their appearance (e.g. color and position) depending on value changes of controller tags.

Points for an existing Poly Line object can be adjusted by selecting **Edit Points** from the context menu.

Line	Rectangle
Ellipse	Polyline

Note:
Bevel for Poly Line and Ellipse is only supported for projects designed for PC targets.

7.3 HMI Controls

Dynamic objects such as buttons, analog numerics, sliders, meters and trend viewers are linked to tags in order to create control and monitoring functions.

The tag associated with an object can be selected and changed from the Tag/Security group on the Home or General ribbon tabs, or from the context menu. Links to tags (controller tags, internal tags or system tags) are selected in the Tag drop-down list.

Objects containing text can be edited directly from the keyboard, either when just created, or when a single selection is done on the text area of the object.

It is also possible to use script code to control the text contents of buttons, texts or text boxes.

The following HMI controls are described:

Action Menu Object	*Circular Meter Object*	*Roller Panel Object*
Alarm Viewer Object	*Chart Object*	*Slider Object*
Analog Numeric Object	*Database Viewer Object*	*Text Object*
Animated GIF	*Digital Clock Object*	*Touch Combo Box Object*
Animated Label Object	*Linear Meter Object*	*Touch List Box Object*
Audit Trail Viewer	*Multi Picture Object*	*Trend Viewer Object*
Button Object	*Picture Object*	

7.3.1 Action Menu Object

The Action menu object is used to display multiple menu groups, each containing its own sets of actions which can be triggered in iX Runtime from the action menu object.

Settings for the Action Menu object are available from the General ribbon tab while the object is selected:

Settings Group

Parameter	Description
Edit Actions	Opens the Edit Actions dialog
View type	Choose between Bar, Expand and Tab
Item Width	Set the width of the items. This is only available for the Bar and Expand View types.
Item Height	Set the height of the items
Orientation	Set the orientation of the items. This is only available for the Bar View type.
Picture Width	Set the width of the pictures for the items
Picture Height	Set the height of the pictures for the items
ScrollBar Width	Set the width of the scroll bar.

Touch Settings Group

Parameter	Description
Scroll Sensitivity	Set the sensitivity of the scroll. Minimum value is 1, maximum value is 10.

Edit Actions Dialog

The Actions tab in the Edit Actions dialog contains the following properties:

Parameter	Description
Text	The text to be displayed for the item
Picture	The picture to be displayed for the item
Make Transparent	Set transparency on the item
Action	Configure the action when the item is clicked

The Menu Groups tab in the Edit Actions dialog contains the following properties:

Parameter	Description
Menu Group Name	The name of the menu group
Picture	The picture to be displayed for the menu group
Make Transparent	Set transparency on the menu group

Note:
Make sure not to use semi-transparent pictures in projects targeted for operator panels. Since these targets do not fully support opacity, semi-transparent pictures may turn out unclear.

Adding a New Action

In the **Edit Actions** dialog select the Actions tab. Click the **Add** button. A new item will be added and placed last in the list. To configure the actions for the item, click the "…" button in the Action column.

Adding a New Menu Group

In the **Edit Actions** dialog select the Menu Groups tab. Click the **Add** button. A new menu group will be added and placed last in the list. The menu group will be displayed in the Actions tab as a new menu group tab.

Action Menu Dynamics

It is also possible to set dynamics, i.e. tag dependent values for the Action Menu properties. These can be found in the Dynamics tab in the ribbon bar.

Related information
Dynamics Ribbon Tab

7.3.2 Alarm Viewer Object

The Alarm Viewer is used to display and manage alarm items, for example indication of out-of-bounds tag values. The viewer consists of a list area and a button area.

Related information
Alarm Management
Alarm Viewer

7.3.3 Analog Numeric Object

The Analog Numeric object displays a box with a numeric or alphanumeric value. The value can be connected to a tag.

Settings for the analog numeric object are available from the General ribbon tab while the object is selected:

Display Format Group

Parameter	Description
Type	Selection of Integer, Decimal, String, Hex or Binary display format
Number of Decimals	Maximum number of decimals to display in the object. Only available when Type = Decimal. The number of decimals will be represented by a "#" symbol on the analog numeric object.
Limit Characters to	Makes it possible to maximize the number of displayed characters. The number of characters will be represented by a "#" symbol on the analog numeric object.
Zero Fill	Fills empty character positions with zeros, to let the maximum number of characters always be displayed

Note:
The **"Limit Characters to"** function, in combination with the string format type is not supported for the Traditional Chinese, Simplified Chinese and Simplified Chinese (Compact) languages.

Prefix/Suffix Group

Parameter	Description
Prefix/Suffix	Text value that will precede/follow the object value

Tag/Security Group

Parameter	Description
Select Tag	Connection to a tag from the tags configuration page
Select Security Groups	Allows limiting access of the object based on security groups. If left empty, the object can be accessed by anyone.
Select Visibility	Configuration of visibility: **Default**: The visibility specified on the General tab of the Security Manager is used. **Disabled**: The object is disabled for users with insufficient access rights **Hidden**: The object is invisible for users with insufficient access rights **Normal**: The object looks normal also for users with insufficient access rights

Format Group

Parameter	Description
Auto Stretch Font	Resizes the font to fit the height of the object. Default setting for the analog numeric object.
Disable Operator Input	Disables input in runtime
Transparent	Makes the object transparent, i.e. makes the frame and background color invisible

Text Alignment Group

Parameter	Description
Horizontal/Vertical	Adjusts the object horizontally or vertically

Validation Group

Parameter	Description
On Input	Input values are validated, based on Lower Limit and Upper Limit, when the box is checked
On Display	Background color changes according to Other Colors on the Format tab if upper or lower limit are violated, when the box is checked
Lower Limit/Upper Limit	Minimum/maximum value that can be entered in runtime when **On Input** is checked, or that will change the background color of the object when **On Display** is checked

Note:
The font size of the analog numeric object is auto-sized when resizing the object frame by default, and is not adjusted in the Font control of the Home ribbon tab. Uncheck **Auto Stretch Font** in the Text group first, and then change the font size on the Home ribbon tab.

7.3.4 Animated GIF

The Animated GIF (Graphics Interchange Format) object allows including a moving graphic picture in the iX Developer project. Animated GIFs can be imported to the project by browsing the PC environment for graphic files. Imported animated GIFs are added to the Project Picture folder in the Component Library.

The Animated GIF can be connected to a tag which controls animation. When the value of the connected tag is 0, then animation is turned off and the picture is displayed static in its initial design. An Animated GIF not connected to a tag runs animated continuously.

Settings for the Animated GIF object are available from the General ribbon tab while the object is selected:

Settings Group

Parameter	Description
Change Picture	Opens a browse dialog for selection of picture
Reset Picture	Restores original height and width to picture
Stretch	Adapts height and width to the selection frame when resizing the picture
Transparent	Makes the object transparent, i.e. makes the frame and background color invisible

Note:
Make sure not to use semi-transparent pictures in projects targeted for operator panels. Since these targets do not fully support opacity, semi-transparent pictures may turn out unclear.

Tag/Security Group

Parameter	Description
Select Tag	Connection to a tag from the tags configuration page
Select Security Groups	Allows limiting access of the object based on security groups. If left empty, the object can be accessed by anyone.
Select Visibility	Configuration of visibility: **Default:** The visibility specified on the General tab of the Security Manager is used. **Disabled:** The object is disabled for users with insufficient access rights **Hidden:** The object is invisible for users with insufficient access rights **Normal:** The object looks normal also for users with insufficient access rights

7.3.5 Animated Label Object

The Animated Label object is used to display an animated text label in iX Runtime.

Settings for the Animated Label object are available from the General ribbon tab while the object is selected:

Note:
Using the animated label object creates a slightly higher load on the CPU.

Parameter	Description
Text	The text to be displayed in the animated label
Animated	If enabled, the text starts to animate
Animation speed	Set the speed for the animation. Minimum value is 1, maximum value is 32767.
Animation Direction	Set the animation direction: From right to left From left to right From right to left to right From left to right to left From down to up From up to down From down to up to down From up to down to up

Animated Label Dynamics

It is possible to set dynamics, i.e. tag dependent values for the Animated Label properties. These can be found in the Dynamics tab in the ribbon bar.

Related information
Dynamics Ribbon Tab

7.3.6   Audit Trail Viewer Object

The Audit Trail Viewer displays information about operator changes that have been logged with the audit trail function.

Related information
Audit Trail
Audit Trail Viewer

7.3.7   Button Object

The Button tool draws a button with a text and/or picture. The button can be configured to perform a specific task in runtime via the Actions ribbon tab or using script. The text on the button object can also be set to depend directly on a tag value using the General control on the Dynamics ribbon tab. If a picture is used, the picture file is added to the Project Pictures folder in the Component Library.

Settings for the button object are available from the General ribbon tab while the object is selected:

Picture Group

Parameter	Description
Picture	Selects a picture to display on the button. Pictures from the Project Pictures folder is available for selection directly. It is also possible to browse the PC environment for a picture.
	Select **No Picture** to remove a previously selected picture.
	Checking the Multi Picture checkbox will make it possible to display different pictures based on the value of a connected tag according to section *Configure Multi Pictures*.
Multi Picture	Multi Picture or single picture displayed on button.
Transparent	Makes the picture transparent

Note:
Make sure not to use semi-transparent pictures in projects targeted for operator panels. Since these targets do not fully support opacity, semi-transparent pictures may turn out unclear.

Configure Multi Pictures

Different pictures can be displayed on the button based on the value of a connected tag, following the steps below:

1. Check the Multi Picture checkbox.
2. Click on the **Picture** control on the Picture group of the General tab.
3. Select a tag to control which picture to display.
4. Click **Add** to add pictures and configure start and end values for each picture. You can browse the PC for pictures, or select pictures from the current project.
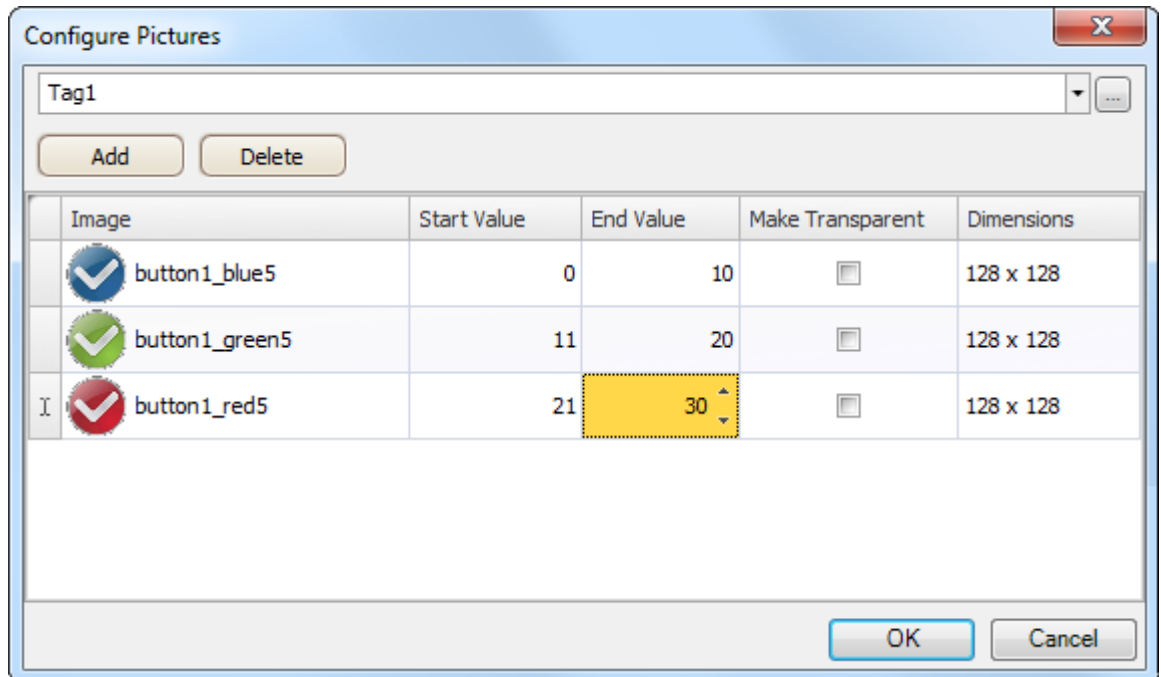
5. Click **OK**.

    The picture on the button will now change according to the value of the selected tag. When the value is out of limits, no picture is displayed.

Picture Alignment Group

Parameter	Description
Vertical Alignment / Horizontal Alignment	Set alignment for the picture

Clicking the small arrow in the lower right corner of the Picture Alignment group displays the picture properties with a number of additional picture settings.

Style Group

Parameter	Description
Style	Predefined graphical styles for the button
Appearance	The appearance of the button, including possibility to make the button rounded and settings for each corner of the button.

Note:

When editing default button style properties, for example corner radius, appearance in runtime may sometimes differ from when designing the project in iX Developer.

Text Group

Parameter	Description
Text	Text displayed on the button
Multiline	Lets the text run over more than one line, using [**Enter**]. Confirm the text entry by pressing [**Ctrl**] + [**Enter**].
Word Wrap	Wraps the text to fit the width of the button
Configure Texts	Links the displayed text to a tag. The Text field value is shown if the selected tag is not within the set limits

Text Alignment Group

Parameter	Description
Horizontal / Vertical	Adjusts the text on the button horizontally or vertically

Visibility Group

Parameter	Description
Transparent	Makes the object transparent

Creating a Button with Momentary Function

In order to create a button with a momentary function, you need to combine a **Set Tag** action for Mouse Down with a **Reset Tag** action for Mouse Up .
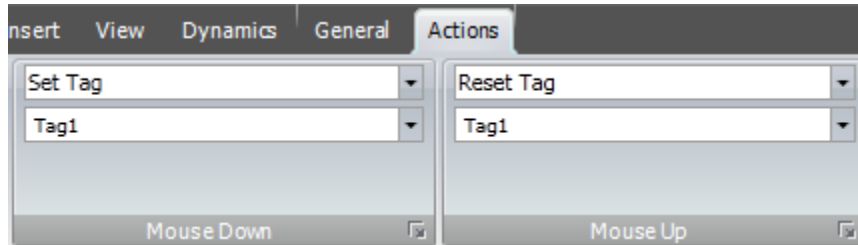
Related information
Actions Ribbon Tab
Script
Component Library

7.3.8 Circular Meter Object

The Circular Meter represents a numeric value, and displays the value of a connected tag.

Different styles can be used to change the graphical profile of the meter, and also if the meter is to be full, half or quarter.
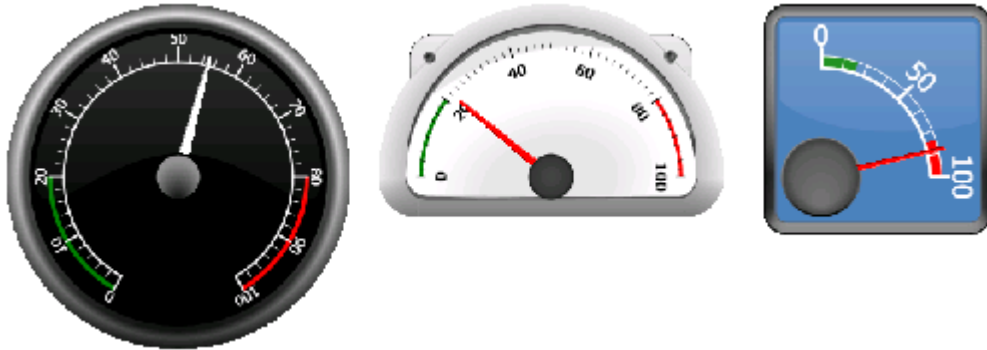
Circular meters in Black Full style, Chrome Half style and Vintage Quarter style

Settings for the meter object are available from the General ribbon tab while the object is selected:

Angular Settings Group

Parameter	Description
Start Angle / End Angle	The angular value is based on a start (0 degrees) at three o'clock
Text Rotation	Select **Parallel** to let the numbers follow the outline of the meter, or **None** to display the numbers aligned horizontally.
Clockwise	Determines rotation orientation, clockwise or counter clockwise.

Region Settings Group

Parameter	Description
Low Region / Medium Region / High Region Min / Max	Define colors for up to 3 regions, and at which values to start and stop each region. Entering 0 for both minimum and maximum values displays no region color.

Value Scale Group

Parameter	Description
Min Value / Max Value	Start / end value of scale
Major Ticks	Number of labeled scale marks on the meter
Minor Ticks	Number of scale ticks between two adjacent major ticks

Style Group

Parameter	Description
Style	Predefined graphical styles for the meter. The smaller styles are intended in particular to allow full visibility in the smaller operator panels.
Appearance	The appearance of the meter; full, half or quarter

Tag/Security Group

Parameter	Description
Select Tag	Connection to a tag from the tags configuration page
Select Security Groups	Allows limiting access of the object based on security groups. If left empty, the object can be accessed by anyone.
Select Visibility	Configuration of visibility: **Default**: The visibility specified on the General tab of the Security Manager is used. **Disabled**: The object is disabled for users with insufficient access rights **Hidden**: The object is invisible for users with insufficient access rights **Normal**: The object looks normal also for users with insufficient access rights

Visibility Group

Parameter	Description
Transparent	Makes the object transparent
Scale	When unchecked, the scale color of the meter is made transparent. Scale color is set from Other Colors on the Format tab.

7.3.9 Chart Object

The chart object is used to display values of connected array tags.

Settings for the chart object are available from the General ribbon tab while the object is selected:

Chart Settings Group

Parameter	Description
Edit Series	Add, remove and edit the current series for this chart
Chart Types	Choose chart type. Available types are: Area Area Horizontal Bar Bar Join Donut Histogram Kagi Line Line Fast Line Horizontal Line Point Pie Points Radar
Header	Set the name that is displayed above the chart
3D View	Set a 3D look and feel for the chart

X Axis Settings Group

Parameter	Description
Visible	Set the visibility of the X axis
Grid Visible	Set the visibility of the X axis grid lines.
Automatic	Automatically calculates the minimum and maximum X axis values. Overrides the Minimum and Maximum properties
Minimum	Set the minimum value of the X axis. If automatic is set, this value is ignored
Maximum	Set the maximum value of the X axis. If automatic is set, this value is ignored

Y1 Axis Settings Group

Parameter	Description
Visible	Set the visibility of the Y1 axis
Grid Visible	Set the visibility of the Y1 axis grid lines.
Automatic	Automatically calculates the minimum and maximum Y1 axis values. Overrides the Minimum and Maximum properties.
Minimum	Set the minimum value of the Y1 axis. If automatic is set, this value is ignored.
Maximum	Set the maximum value of the Y1 axis. If automatic is set, this value is ignored. When using charts with visible labels on curves the maximum value is extended to make room for label.

Y2 Axis Settings Group

Parameter	Description
Visible	Set the visibility of the Y2 axis
Grid Visible	Set the visibility of the Y2 axis grid lines.
Automatic	Automatically calculates the minimum and maximum Y2 axis values. Overrides the Minimum and Maximum properties.
Minimum	Set the minimum value of the Y2 axis. If automatic is set, this value is ignored.
Maximum	Set the maximum value of the Y2 axis. If automatic is set, this value is ignored. When using charts with visible labels on curves the maximum value is extended to make room for label.

Style Group

Parameter	Description
Style	Set the style of the chart

Visibility Group

Parameter	Description
Scale	Set the visibility of the scale
Transparent	Make the chart transparent
Header	Set the header visibility
Legend	Set the legend visibility

The Edit Series Dialog

Clicking the **Edit Series** control in the Chart Settings Group will open the Edit Series Dialog. The Edit Series Dialog contains the following settings:

Parameter	Description
Name	The name of the Series, as shown in the legend in the chart.
X Tag	The X Tag values. The tag must be an array tag. The number of elements shown will be the lowest common denominator of the X Tag array size and the Y Tag array size.
Y Tag	Optional: The Y Tag values for the chart. The tag must be an array tag. If this value is not chosen, the X Tag Values will still be visible and the Y Axis will be set automatically. The number of elements shown will be the lowest common denominator of the X Tag array size and the Y Tag array size.
Size	Optional: The number of elements in the tag arrays. This property overrides the sizes of the X Tag array size and Y Tag array size. If this value is not set, the lowest common denominator of the X Tags array size and the Y Tags array size will be chosen as size.
Labels	Set a label for each point in the chart. There is one label for each available tag value.
Color	Choose the color of the series.
Thickness	Set the thickness of the series. This applies for the chart types Line, Line Fast, Line Horizontal and Line Point.
Axis Type	Select the Y Axis that this series will use.

Note:
Chart objects with large arrays where values are frequently changing may affect performance.

Related information
Array Tags

7.3.10 Database Viewer Object

The Database Viewer can be used to display database contents in runtime.

For operator panel projects, only databases included in the current project are available. For PC targets, it is possible to access databases from anywhere in the network environment.

Settings for the database viewer object are available from the General ribbon tab while the object is selected:

Settings Group

Parameter	Description
Show Controls	When the box is checked, the user is able to select database and database table in runtime
Database	Path to database to be loaded automatically in runtime*
Table Name	The table in the selected database to be loaded automatically in runtime*
Browse Data Table	Opens a browse dialog for selection of database and database table in design time. The path and table name will be added to the **Database** and **Table Name** fields.

* If the entered database or table does not exist or is not valid, an empty database viewer is displayed in runtime.

Note:
Simulation of the object on the development PC is supported only for PC targets.

Related information
Database

7.3.11 Digital Clock Object

The digital clock displays date, time and/or day of week.

The time zone is selected on the System ribbon tab and is transferred to the target when the project is downloaded to it. If no time zone is selected, the time zone that has been selected for the panel or PC will be retained.

The date and time display format (for example using AM/PM for time) is based on the operating system settings. For operator panel projects, this format may be altered by selecting another region on the System ribbon tab.

Display Format Group

Parameter	Description
DateAndTime/ Date/Time	Selection of date and time components
Show Day of Week	When checked, day of week is displayed
Show Seconds	When checked, seconds are displayed

7.3.12   Linear Meter Object

The Linear Meter represents a numeric value, and displays the value of a connected tag.

Different styles can be used to change the graphical profile of the meter, and also if the meter is to be horizontal or vertical.

Linear meters in Classic Horizontal style and Eclipse Vertical style

Settings for the meter object are available from the General ribbon tab while the object is selected:

Value Scale Group

Parameter	Description
Min Value / Max Value	Start / end value of scale
Major Ticks	Number of labeled scale marks on the meter
Minor Ticks	Number of scale ticks between two adjacent major ticks

Style Group

Parameter	Description
Style	Predefined graphical styles for the meter. The smaller styles are intended in particular to allow full visibility in the smaller operator panels.
Appearance	The appearance of the meter; horizontal or vertical

Tag/Security Group

Parameter	Description
Select Tag	Connection to a tag from the tags configuration page
Select Security Groups	Allows limiting access of the object based on security groups. If left empty, the object can be accessed by anyone.
Select Visibility	Configuration of visibility: **Default**: The visibility specified on the General tab of the Security Manager is used. **Disabled**: The object is disabled for users with insufficient access rights **Hidden**: The object is invisible for users with insufficient access rights **Normal**: The object looks normal also for users with insufficient access rights

Visibility Group

Parameter	Description
Transparent	Makes the object transparent; only indicator bar and scale will be visible
Scale	When unchecked, the scale color of the meter is made transparent. Scale color and indicator color are set from Other Colors on the Format tab.

7.3.13   Multi Picture Object

The Multi Picture object allows using a set of pictures as an object. The pictures change in runtime depending on intervals of a tag value.

Pictures can be imported to the project by browsing the hard disk for graphic files. Imported pictures are added to the Project Pictures folder in the Component Library.

Settings for the Multi Picture object are available from the General ribbon tab while the object is selected:

Settings Group

Parameter	Description
Configure Pictures	Configuration of start/end values of intervals and selection of pictures for each interval. Pictures can be selected from the Project Pictures folder or from the hard disk. Transparency can also be set. Original picture size is displayed. The **Add/Delete** buttons are used to add and remove items to/from the object.
Reset Picture	Restores original height and width to picture.
Stretch	Adapts height and width to the selection frame when resizing the picture.
Lock Aspect Ratio	Maintains the height/width ratio of the object.

Note:
Make sure not to use semi-transparent pictures in projects targeted for operator panels. Since these targets do not fully support opacity, semi-transparent pictures may turn out unclear.

Related information
Component Library

7.3.14   Picture Object

The Picture object allows using a picture as an object. Pictures can be imported to the project by browsing the PC environment for graphic files. Imported pictures are added to the Project Pictures folder in the Component Library.

The displayed picture can also be dynamic; depending of a set of pictures that have been linked to intervals in a tag value.

Settings for the picture object are available from the General ribbon tab while the object is selected:

Settings Group

Parameter	Description
Change Picture	Opens a browse dialog for selection of picture
Reset Picture	Restores original height and width to picture
Stretch	Adapts height and width to the selection frame when resizing the picture
Transparent	Makes the object transparent, i.e. makes the frame and background color invisible

Note:
Make sure not to use semi-transparent pictures in projects targeted for operator panels. Since these targets do not fully support opacity, semi-transparent pictures may turn out unclear.

Related information
Component Library

7.3.15   Roller Panel Object

The Roller Panel object is used to display predefined texts for certain tag values. Settings for the Roller Panel object are available from the General ribbon tab while the object is selected:

Settings Group

Parameter	Description
Title	Set the title for the Roller Panel
Title Visible	Determines if the title is visible or not
Loop Scroll	Determines if the values will loop around or not
Disable Operator Input	Determines if the Roller Panel is editable or not

Touch Settings Group

Parameter	Description
Scroll Sensitivity	Set the sensitivity of the scroll
Scroll Friction	Set the friction of the scroll

Text Group

Parameter	Description
Configure Texts	Opens the Configure Texts dialog

Configure Text Dialog

The Configure Texts dialog contains the following properties:

Parameter	Description
Text	The text to be displayed for the item
Start Value	The starting value for when this item should be selected
End Value	The ending value for when this item should be selected

Roller Panel Dynamics

It is also possible to set dynamics, i.e. tag dependent values for the Roller Panel properties. These can be found in the Dynamics tab in the ribbon bar.

Related information
Dynamics Ribbon Tab

7.3.16   Slider Object

The Slider can be used to adjust a numeric value. The slider sets a value to a connected tag. If the tag value changes for other reasons, the slider control handle will show the actual value in runtime.

Sliders in Chrome and Eclipse Horizontal styles, and Vintage Vertical style

Settings for the slider object are available from the General ribbon tab while the object is selected:

Value Scale Group

Parameter	Description
Min Value / Max Value	Start / end value of scale
Major Ticks	Number of labeled scale marks on the slider
Minor Ticks	Number of scale ticks between two adjacent major ticks

Style Group

Parameter	Description
Style	Predefined graphical styles for the slider. The smaller styles are intended in particular to allow full visibility in the smaller operator panels.
Appearance	The appearance of the slider; horizontal or vertical

Tag/Security Group

Parameter	Description
Select Tag	Connection to a tag from the tags configuration page
Select Security Groups	Allows limiting access of the object based on security groups. If left empty, the object can be accessed by anyone.
Select Visibility	Configuration of visibility: **Default**: The visibility specified on the General tab of the Security Manager is used. **Disabled**: The object is disabled for users with insufficient access rights **Hidden**: The object is invisible for users with insufficient access rights **Normal**: The object looks normal also for users with insufficient access rights

Visibility Group

Parameter	Description
Transparent	Makes the object transparent
Scale	When unchecked, the scale color of the slider is made transparent. Scale color and indicator color (color of the handle) are set from Other Colors on the Format tab.

7.3.17   Text Object

The Text object is used to display read-only information for the operator. The displayed text can be static, or linked to intervals in a tag value that contain different strings. The text object can also be set to depend directly on a tag value using the General control on the Dynamics ribbon tab.

Settings for the text object are available from the General ribbon tab while the object is selected:

Text Group

Parameter	Description
Text	Text that the object displays
Multiline	Lets the text run over more than one line, using [**Enter**]. Confirm the text entry by pressing [**Ctrl**] + [**Enter**].
Word Wrap	Wraps the text to fit the width of the object. Not available if Autosize is selected.
Configure Texts	Links the displayed text to a tag. The Text field value is shown if the selected tag is not within the set limits.
Auto Stretch Font	Resizes the font to fit the height of the object. Not available if Autosize is selected.
Auto Size	Resizes the object according to the size of the default text string length and font size. Default setting for the text object.

Text Alignment Group

Parameter	Description
Horizontal/Vertical	Adjusts the text in the object horizontally or vertically. Not available if Autosize is selected.

7.3.18   Touch Combo Box Object

The Touch Combo Box object is used to create text selections from a drop-down list.

Settings for the Touch Combo Box object are available from the General ribbon tab while the object is selected:

Settings Group

Parameter	Description
Auto Size Item Height	The Touch Combo Box will ignore the Item Height value and size the values automatically
Item Height	Set the height of the individual items in the Touch Combo Box
ArrowBox Width	Set the width of the Arrow Box
ScrollBar Width	Set the width of the scroll bar

Text Group

Parameter	Description
Configure Texts	Opens the Configure Texts dialog

Configure Text Dialog

The Configure Texts dialog contains the following properties:

Parameter	Description
Text	The text to be displayed for the item
Start Value	The starting value for when this item should be selected
End Value	The ending value for when this item should be selected

Touch Settings Group

Parameter	Description
Scroll Sensitivity	Set the sensitivity of the scrolling

Touch Combo Box Dynamics

It is also possible to set dynamics, i.e. tag dependent values for the Touch Combo Box properties. These can be found in the Dynamics tab in the ribbon bar.

Related information
Dynamics Ribbon Tab

7.3.19 Touch List Box Object

The Touch List Box object is used to handle a list of predefined texts.

Settings for the Touch List Box object are available from the General ribbon tab while the object is selected:

Settings Group

Parameter	Description
Separator Visible	Show or hide a separator between the items in the Touch List Box
Item Height	Set the height of the individual items in the Touch List Box
ScrollBar Width	Set the width of the scroll bar

Touch Settings Group

Parameter	Description
Scroll Sensitivity	Set the sensitivity of the scrolling

Text Group

Parameter	Description
Configure Texts	Opens the Configure Texts dialog

Configure Texts Dialog

The Configure Texts dialog contains the following properties:

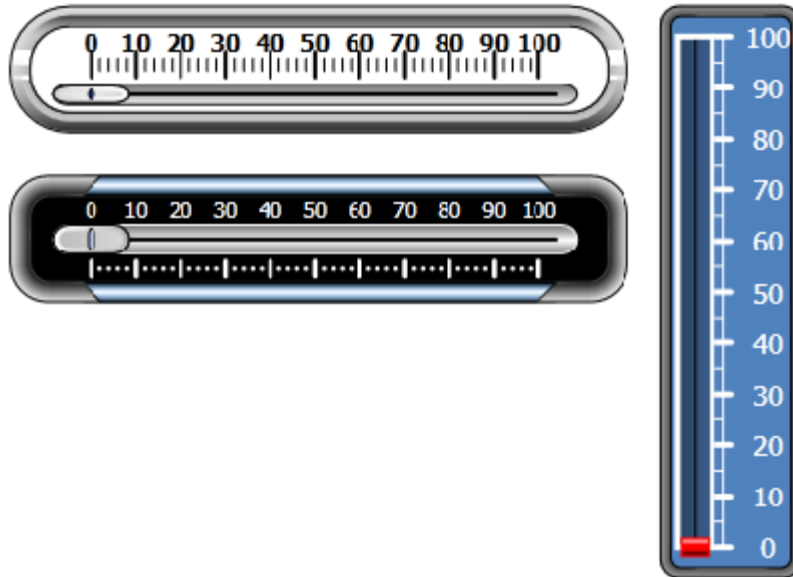| Parameter	Description
Text	The text to be displayed for the item
Start Value	The starting value for when this item should be selected
End Value	The ending value for when this item should be selected

Touch List Box dynamics

It is also possible to set dynamics, i.e. tag dependent values for the Touch List Box properties. These can be found in the Dynamics tab in the ribbon bar.

Related information
Dynamics Ribbon Tab

7.3.20   Trend Viewer Object

Trend viewers are used to display values of connected tags. Trend viewer curves represent tag values over time. A trend viewer object can show any number of trend viewer curves.

Note:
A large number of trend viewer pens and short sampling intervals may affect communication performance.

Related information
Trend Viewer
Curves Group
Communication Performance

7.4 Media Controls

The media controls are used to display media files, pdf files and web pages in runtime. The following media controls are described:

Media Player Object	*PDF Viewer Object*	*Web Browser Object*

7.4.1   Media Player Object

Media files can be added to the media player object, which allows selecting, starting and stopping the files in runtime. For operator panels, files in .mpeg1 and .wmv format are supported. The added files are added to the Component Library.

Note:
Microsoft Media Player of version 10 or later is required.

Browse Folders Group

Parameter	Description
Project Files	Default option for browsing. The project files are listed in the Component Library. At least one of the folders has to be chosen.
External Memory Card	Select this option for browsing if project files are placed on an external memory card. At least one of the folders has to be chosen.

Settings Group

Parameter	Description
Media Source	Select a media file to add to the media player, from the project files folder or by browsing the PC

Note:
The object is not supported on Panel TxA targets. Simulation of the object on the development PC is supported only for PC targets.

Scripting to the Media Player Object

When scripting to the Media Player object, the name of the current media file has to be stated, for example:

MediaPlayer1.Source = "Bear.wmv";

Prior to running the script, the media file has to be added manually to the Project Files folder. When the script is executed, the file will be copied to the output directory.

Events for the media player can be used, for example:

MediaPlayer1.Play();

Related information
Component Library

7.4.2 PDF Viewer Object

Pdf files can be displayed in runtime using the PDF viewer object. The added files are available from the Component Library.

Note:
Adobe Reader of version 9 or 9.5 is required.

Browse Folders Group

Parameter	Description
Project Files	Default option for browsing. The project files are listed in the Component Library. At least one of the folders has to be chosen.
External Memory Card	Select this option for browsing if project files are placed on an external memory card. At least one of the folders has to be chosen.

Settings Group

Parameter	Description
PDF File	Select a PDF file to add to the PDF viewer, from the project files folder or by browsing the PC
Buttons	Select which buttons to make available in runtime

In an operator panel, all PDF files in the project files folder will be available for selection in runtime, if the Open button is checked.

Note:
When using a PDF viewer object in a popup screen, only the default style for the popup screen is supported.

Note:
The object is not supported on Panel TxA targets. Simulation of the object on the development PC is supported only for PC targets.

Related information
Component Library
Popup Screen
Adobe Reader not Recognized

7.4.3 Web Browser Object

Web pages can be displayed using the web browser. It is possible to set up the web browser with regular web browsing functionality, with selection of web addresses in a list or input via the on screen keyboard. It is also possible to allow showing certain defined web pages only, or a local htm/html file.

Note:
Microsoft Internet Explorer of version 9 or later is required.

Settings Group

Settings for the Web Browser object are available from the General ribbon tab while the Web Browser object is selected:

Parameter	Description
Address	The default web page of the web browser object.
Home	The web page that will be displayed when the Home button is pressed.
Favorites	Addition of web pages that will be available for selection from the drop-down address field in runtime.
Address Read Only	Checking the box disables entering text in the address field
Address Field	Unchecking the box removes the address field in runtime
Go Button	Unchecking the box removes the Go button in runtime
Home Button	Unchecking the box removes the Home button in runtime
Back Button/Forward Buttons	Unchecking the box removes the navigation buttons in runtime

Referring to a local htm/html file can be useful for example when the operator panel is not connected to Internet. The local file has to be located in the Project Files folder of the current project, and is addressed in the following format: "File://filename.htm/html" where "filename.htm/html" represents the actual file", e.g. *File://ReadMe.htm*.

Web Browser Limitations

The web browser object functionality is limited under some circumstances:

Function	Limitation description	PC targets	Operator Panel targets
Web browser object		Supported	Not supported on Panel TxA targets.
Popup windows	Opening popup windows from the web browser in runtime	Not supported	Not supported
Web browser in a popup screen	Other styles than the default style for the popup screen	Not supported	Not supported
Simulation of the web browser	Simulation on the development PC	Supported	Not supported
Support of some web pages	Access of some web pages from within the web browser.	Supported	May not be supported. If the web browser does not behave as expected, it is recommended to use the standard Internet Explorer, available in the Operator panel, instead. The standard Internet Explorer can be started using the **Run** action and executing the iesample.exe file.

Related information
Popup Screen

7.5 Special Controls

The special controls are used to create attractive navigation facilities in runtime.

Navigation List Box Object	Screen Carousel Object

7.5.1   Navigation List Box Object

Screen navigation using the Navigation List Box object is performed via thumbnails, and scripting can be used for advanced functionality.

Note:
The object is supported only for Panel TxC and PC targets.

General Group

Parameter	Description
Orientation	Select horizontal or vertical orientation for runtime scrolling.
Scroll Tolerance	Defines tolerance of pressing the touch screen (or scrolling with the cursor) slightly outside of a navigation list box item in runtime before the navigation is actually carried out.

Item Settings Group

Parameter	Description
Configure Items	Definition of items included in the navigation list box object; see below.
Height/Width	Size of the items in the navigation list box.
Margin	Margin between navigation list box items.
Border Width	Width of navigation list box item borders.
Corner Radius	Corner radius of navigation list box items.
Show Text	Selection of whether to show text below navigation list box items.

The individual items in the navigation list box object are handled by clicking **Configure Items**. Use the **Add** and **Delete** buttons to add and remove items to/from the navigation list box.
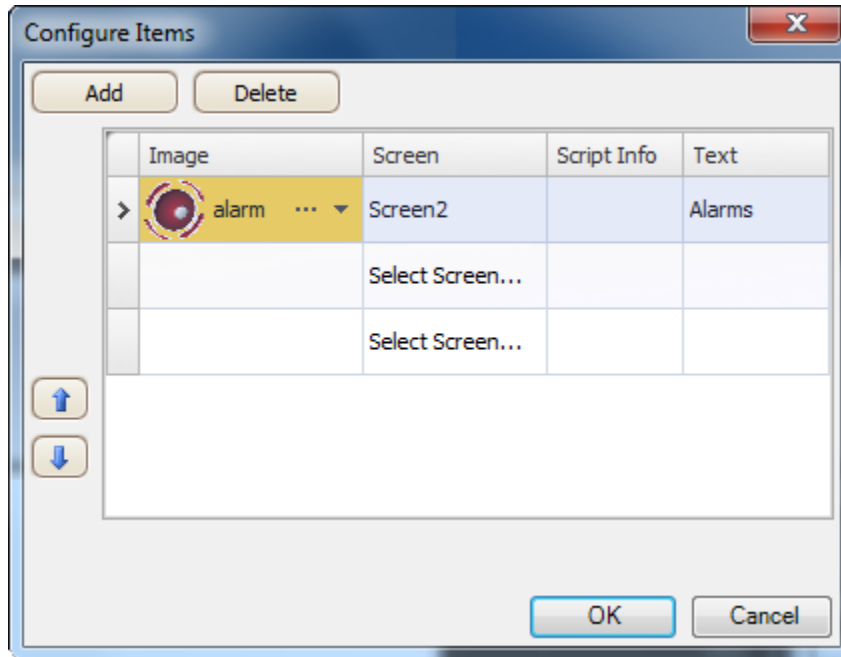
Parameter	Description
Picture	Selection of picture by browsing the PC environment or among the project files.
Screen	Selection of screen to navigate to.
Script Info	Optional text to serve as a reminder if scripts are used.
Text	Text displayed under each navigation list box item. Only visible if Show Text is checked under Item Settings.

Use the up and down arrows to facilitate reorganizing of navigation list box items.

Related information
Script

7.5.2 Screen Carousel Object

Screen navigation using the Screen Carousel object is performed via thumbnails in a 3D view. The screen carousel can be configured to rotate automatically, or actions can be configured to control screen carousel behavior.

Note:
The object is supported only for Panel TxC and PC targets.

General Group

Parameter	Description
Number of Visible Items	Maximum number of visible items.
Scroll Duration (ms)	Time used for an item to move to the next position in the carousel.
Use Reflection	Adds a reflecting layout to the object.
Auto Scroll	Makes the object rotate automatically.

Item Settings Group

Parameter	Description
Configure Items	Definition of items included in the screen carousel object; see below.
Height/Width	Size of the items in the screen carousel.
Margin	Margin between screen carousel items.
Border Width	Width of screen carousel item borders.
Corner Radius	Corner radius of screen carousel items.
Show Text	Selection of whether to show text below screen carousel items.

The individual items in the screen carousel object are handled by clicking **Configure Items**. Use the **Add** and **Delete** buttons to add and remove items to/from the screen carousel.
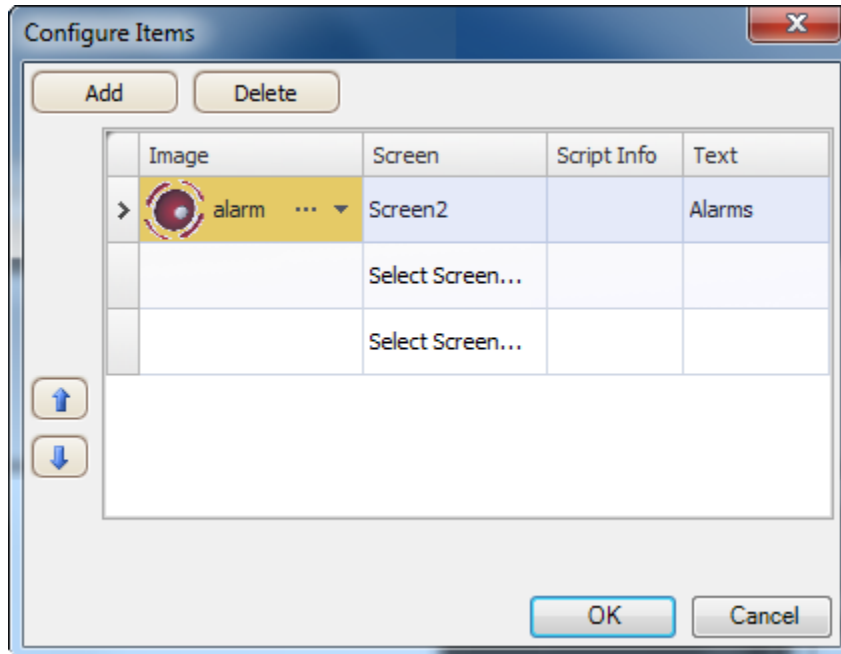
Parameter	Description
Picture	Selection of picture by browsing the PC environment or among the project files.
Screen	Selection of screen to navigate to.
Script Info	Optional text to serve as a reminder if scripts are used.
Text	Text displayed under each screen carousel item. Only visible if Show Text is checked under Item Settings.

Use the up and down arrows to facilitate reorganizing of screen carousel items.

Related information
Actions Ribbon Tab

7.6   Debug Tools

Debug tools are used for trouble shooting and monitoring.

Alarm Distributor Viewer Object	

7.6.1 Alarm Distributor Viewer Object

The Alarm Distributor Viewer is a debug tool, used to display information from the Alarm Distributor function.

Related information
Alarm Management
Alarm Distributor Viewer

7.7   Windows Controls

The following Windows controls are described:

Check Box Object	*Progress Bar Object*
Combo Box Object	*Radio Button Object*
Group Box Object	*Text Box Object*
List Box Object	

7.7.1 Check Box Object

The Check Box object controls a digital tag value. The connected tag gets the value 1 when the box is checked and the value 0 when the box is unchecked.

The appearance of the text can be adjusted in the Font group of the Home ribbon tab.

Note:
Changing font family is only supported for projects designed for panel TxC and PC targets.

Settings for the check box are available from the General ribbon tab while the object is selected:

Text Group

Parameter	Description
Text	Text that the object displays
Multiline	Lets the text run over more than one line, using [**Enter**]. Confirm the text entry by pressing [**Ctrl**] + [**Enter**].
Word Wrap	Wraps the text to fit the width of the object. Not available if Autosize is selected.
Auto Stretch Font	Resizes the font to fit the height of the object. Not available if Autosize is selected.
Auto Size	Resizes the object according to the size of the default text string length and font size.

Text Alignment Group

Parameter	Description
Horizontal/Vertical	Adjusts the text in the text box horizontally or vertically. Not available if Autosize is selected.

7.7.2 Combo Box Object

A Combo Box can be used to create selections from a drop-down list. When a keyboard is connected to the operator panel it is also possible to make selections from the list by typing.

Settings for the combo box are available from the General ribbon tab while the object is selected:

Text Group

Click **Configure Texts** to enter texts for different values.
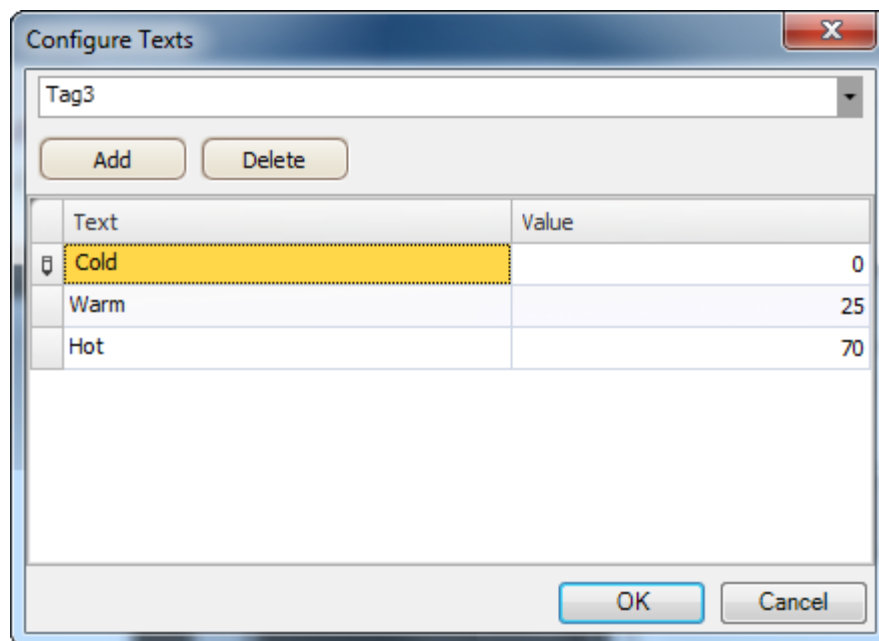
Select a tag, and then click **Add** until there are enough rows in the table. Enter texts and start values.

It is also possible to use script code to control the contents of a combo box.

Example

The following script code relies on the existence of a combo box (ComboBox1) and a set of predefined recipes ("Book" and "TV").

```
void Screen1_Opened(System.Object sender, System.EventArgs e)
{
    ComboBox1.Items.Clear();
    ComboBox1.Items.Add("Book");
    ComboBox1.Items.Add("TV");
}


void ComboBox1_SelectionChanged(System.Object sender, System.EventArgs e)
{
    string selectedItem = ComboBox1.SelectedItem as string;
    Globals.Recipe1.LoadRecipe(selectedItem);
}
```

When the screen opens, the script clears the box and writes the recipe name strings in the box. Selection from the combo box triggers loading of a recipe.

Related information
Script
Recipe Management

7.7.3 Group Box Object

The Group Box is used to draw a labeled frame around a group of objects.

7.7.4 List Box Object

The List Box handles a list of predefined texts.

Text Group

Click **Configure Texts** to enter texts for different values.
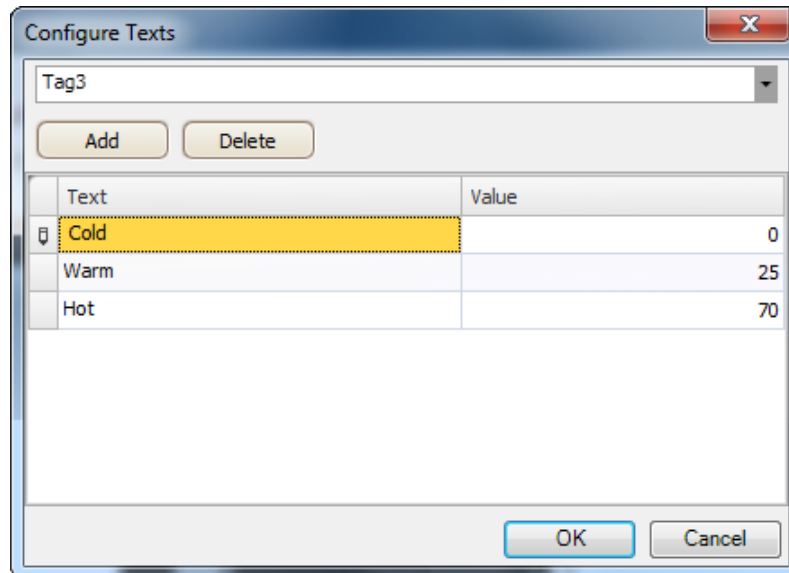
Select a tag, and then click **Add** until there are enough rows in the table. Enter texts and start values.

It is also possible to use script code to control the contents of a list box.

Example

The following script code includes two objects on Screen1, a button (Button1) and a list box (ListBox1).

```
public partial class Screen1
{
    int count=0;
    void Button1_Click(System.Object sender, System.EventArgs e)
    {
        ListBox1.Items.Add("Count " + count.ToString());
        count++;
    }
}
```

When the button is clicked, the script is triggered and a string is written to the list box and the value of an internal counter (count) is incremented.

Related information
Script

7.7.5　Progress Bar Object

The Progress Bar displays a meter without numeric scaling, representing an analog value. The value can be connected to a tag.

The other properties, such as orientation, minimum and maximum values, and meter color (foreground property) are set in the Property grid.

Related information
Property Grid

7.7.6 Radio Button Object

A Radio Button controls the value of a connected digital tag.

The appearance of the text can be adjusted in the Font group of the Home ribbon tab.

Settings for the radio button are available from the General ribbon tab while the object is selected:

Text Group

Parameter	Description
Text	Text that the object displays
Multiline	Lets the text run over more than one line, using [**Enter**]. Confirm the text entry by pressing [**Ctrl**] + [**Enter**].
Word Wrap	Wraps the text to fit the width of the object. Not available if Autosize is selected.
Auto Stretch Font	Resizes the font to fit the height of the object. Not available if Autosize is selected.
Auto Size	Resizes the object according to the size of the default text string length and font size.

Text Alignment Group

Parameter	Description
Horizontal / Vertical	Adjusts the radio button text horizontally or vertically. Not available if Autosize is selected.

It is possible to use script code to make a group of radio buttons enable combined control of a tag value. Exactly one of the radio buttons is active (set to 1) at any time. The following example includes 3 radio buttons and a tag of int16 type.

```csharp
public partial class Screen1
{

    void Screen1_Opened(System.Object sender, System.EventArgs e)
    {
        Globals.Tags.Tag1.ValueChange += Tag1_ValueChanged;
        SetRadioButtonState(Globals.Tags.Tag1.Value);
    }

    void Screen1_Closed(System.Object sender, System.EventArgs e)
    {
        Globals.Tags.Tag1.ValueChange -= Tag1_ValueChanged;
    }

    private void Tag1_ValueChanged(System.Object sender, System.EventArgs e)
    {
        SetRadioButtonState(Globals.Tags.Tag1.Value);
    }

    private void SetRadioButtonState(int Value)
    {
        RadioButton1.Checked = (Value == 1);
        RadioButton2.Checked = (Value == 2);
        RadioButton3.Checked = (Value == 3);
    }

    void RadioButton1_Click(System.Object sender, System.EventArgs e)
    {
        Globals.Tags.Tag1.Value = 1;
    }

    void RadioButton2_Click(System.Object sender, System.EventArgs e)
    {
        Globals.Tags.Tag1.Value = 2;
    }

    void RadioButton3_Click(System.Object sender, System.EventArgs e)
    {
        Globals.Tags.Tag1.Value = 3;
    }

}
```

Radio buttons are grouped based on which screen they belong to.

Related information
Script

7.7.7 Text Box Object

The Text Box is used to include a box with multiple rows of text.

By default, the text can be modified when clicking on the box in runtime. The entered text is not saved in runtime; when the screen changes the value will be reset. The text box can also be configured as read-only.

Settings for the text box object are available from the General ribbon tab while the object is selected:

Text Group

Parameter	Description
Text	Text that the object displays
Multiline	Lets the text run over more than one line, using **[Enter]**. Confirm the text entry by pressing **[Ctrl]** + **[Enter]**.
Word Wrap	Wraps the text to fit the width of the object
Read Only	Disables input in runtime

Text Alignment Group

Parameter	Description
Horizontal / Vertical	Adjusts the radio button text horizontally or vertically

Visibility Group

Parameter	Description
Transparent	Makes the object transparent, i.e. everything but the text will be invisible

7.8   Additional Controls

iX Developer allows using and creating third party controls in order to enhance application functionality and additional customization. This section describes different technologies and includes configuration examples. Sample projects are available at the Beijer Electronics web site.

To understand and use all the information in this document, .Net development skills are required.

Related information
Referenced Assemblies
Error Related to Third Party Controls

7.8.1   Target Platform

Different technologies are used for third party controls depending on the target platform for the iX Developer application. The target can be either PC or Windows CE.

Windows CE has no support for vector graphic (WPF) and only uses .Net Compact Framework which is a subset or the .Net Framework used on a PC. Windows CE does not natively support GDI+, so GDI+ related functionality was removed from .Net Compact Framework.

PC Target

Two different technologies can be used for a PC target:

- Standard Windows forms and GDI+
- WPF (Windows Presentation Foundation)

WPF uses vector graphics, and the appearance of the control is described in XAML. Since iX Developer is a WPF application, it is recommended to use WPF when developing customized controls or user controls for a PC target. Controls developed in WPF can bind to a tag value in iX Developer, in opposite to Windows forms controls, that cannot be bound to tag values.

Windows CE Target

Windows CE only uses the .Net Compact Framework (a subset or the .Net Framework used on a PC), and does not support vector graphics (WPF). Windows CE does not natively support GDI+, so GDI+ related functionality was removed from the .Net Compact Framework.

Limitations

Some of the limitations regarding third party controls are listed below:

- Control Designers (a designer class that can extend design time support) are currently not supported.
- TypeConverters in a separate design dll are not supported.
- Complex property editing in the property grid is not supported. All complex properties have to be set up in script.
- .Net Compact Framework controls can include design dll and so called AssmetaData dll to handle attributes that are not supported in Windows CE. Currently this is not supported by iX Developer. Because of this, it is important to always test the code on the target platform.
- The Script Editor allows scripting against properties and methods that are not supported in Windows CE. Because of this, it is important to always test the code on the target platform.

7.8.2 Adding Controls to the iX Developer Toolbox

Third party controls can be added to the Objects toolbox in iX Developer, following the steps below:

1. Select the **Objects** group on the **Home** ribbon tab, and fully expand the Objects toolbox by clicking the lower right arrow.
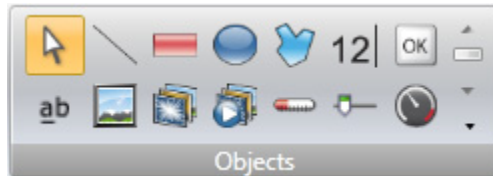
2. Click **Add Control**.

3. Select controls to add among the default controls, or click **Browse** to add customized controls.
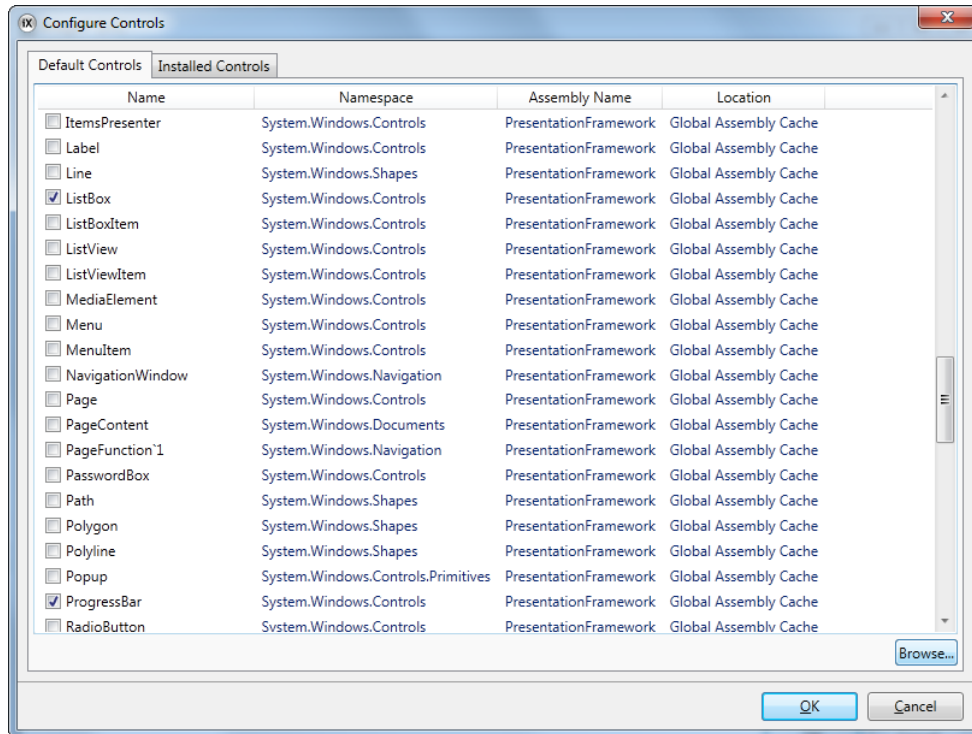
4. Click **OK**.

The added controls are now available under **Additional Controls** in the Objects toolbox.

7.8.3 Default Controls and Installed Controls

Default controls include controls added by the user and the .Net 4 controls installed with the .Net Framework.

Installed controls include all controls that are installed in the GAC (Global Assembly Cache) on your computer.

Note:
Third party controls that are used in a project are not copied to the project folder. This means that it is not possible to open a project with third party controls on another PC without installing the controls. But the application will work in runtime on another target, since references are copied to the output folder when building the project.

7.9   WPF Controls

WPF (Windows Presentation Foundation) uses vector graphics, and the appearance of the control is described in XAML. Since iX Developer is a WPF application, it is recommended to use WPF when developing customized controls or user controls for a PC target. Controls developed in WPF can bind to a tag value in iX Developer.

User controls and custom controls are supported in WPF.

7.9.1   WPF User Controls

A WPF user control can be described as a composition of different user interface controls. Creating a WPF user control is similar to creating a window:

- You have a XAML file and C# class file for a user control.
- The class file extends the user control class, adding additional behavior and properties.
- The XAML file encapsulates the composing controls; styles, templates, animations and whatever necessary for "Look & Feel".

Since the WPF user control is a just composition, it is really easy to create. It does not require a lot of WPF UI model knowledge.

7.9.2   WPF Custom Controls

WPF custom controls are more flexible, but are more complicated than a user control, and require a profound understanding of the WPF user interface model.

- A number of certain user interface controls, such as button, progress bar or speedometer has to be extended.
- The appearance of the custom control has to be defined in XAML, as the custom control itself has no look.

Most of the controls in iX Developer are custom controls, which makes it possible to restyle them to various different layouts without changing the code files; just the XAML.

A rounded meter in different styles

7.9.3   Creating a WPF User Control with Tag Connection

The following example describes how to create a WPF user control that can be connected to a tag.

The complete code is included at the end of the example.

1. Start Visual Studio to create a new project, and select **WPF User Control Library**.
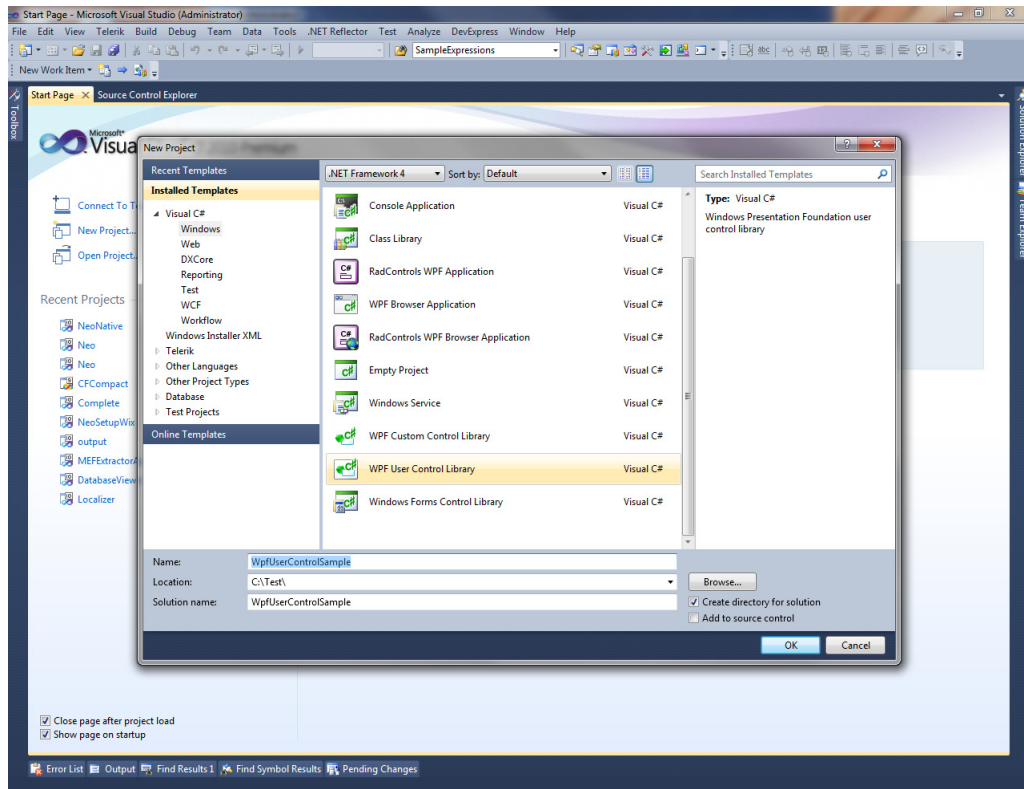
2. Add [DefaultProperty("Value")] to the class, to define which property the tag should set when then value is set.

3. Add a dependency property with same name as the attribute above:
static readonly DependencyProperty ValueProperty;

4. Add a static constructor and register to the dependency property.

5. Create a Value property of type object.

6. Add a TextBox to the user control.

7. Add a binding to the TextProperty and bind to the Value Property.

```
<TextBox Text="{Binding Value, ElementName=userControl, FallbackValue=0}"
Name="textBlock1" Background="#FFF7EFEF" TextAlignment="Center" />
```

8. Remember to change ElementName to the name of your control.

9. Compile and test by adding the control to the iX Developer toolbox.

Note:

When an update is made, the existing control must be updated under C:\Users\Public\Documents\Beijer Electronics AB\iX Developer\Thirdparty\.

Example Code

```csharp
using System;
using System.Collections.Generic;
using System.Windows;
using System.Windows.Controls;

namespace WpfUserControlSample
{
    /// <summary>
    /// Interaction logic for UserControl1.xaml
    /// </summary>
    [DefaultProperty("Value")]
    public partial class SampleUserControl : UserControl
    {
        public static readonly DependencyProperty ValueProperty;
        static SampleUserControl()
        {
            FrameworkPropertyMetadata frameworkPropertyMetadata = new
            FrameworkPropertyMetadata("0", FrameworkPropertyMetadataOptions.Journal |
            FrameworkPropertyMetadataOptions.BindsTwoWayByDefault);

            ValueProperty = DependencyProperty.Register("Value", typeof(object),
            typeof(SampleUserControl), frameworkPropertyMetadata);
        }

        public SampleUserControl()
        {
            InitializeComponent();
        }

        public object Value
        {
            get { return GetValue(ValueProperty); }
            set { SetValue(ValueProperty, value); }
        }
    }
}
```

7.9.4 Creating a Windows Forms User Control for a PC Target

The following example describes how to create a Windows Forms user control designated for a PC target.

1. Start Visual Studio to create a new project, and select **Windows Forms Control Library**.
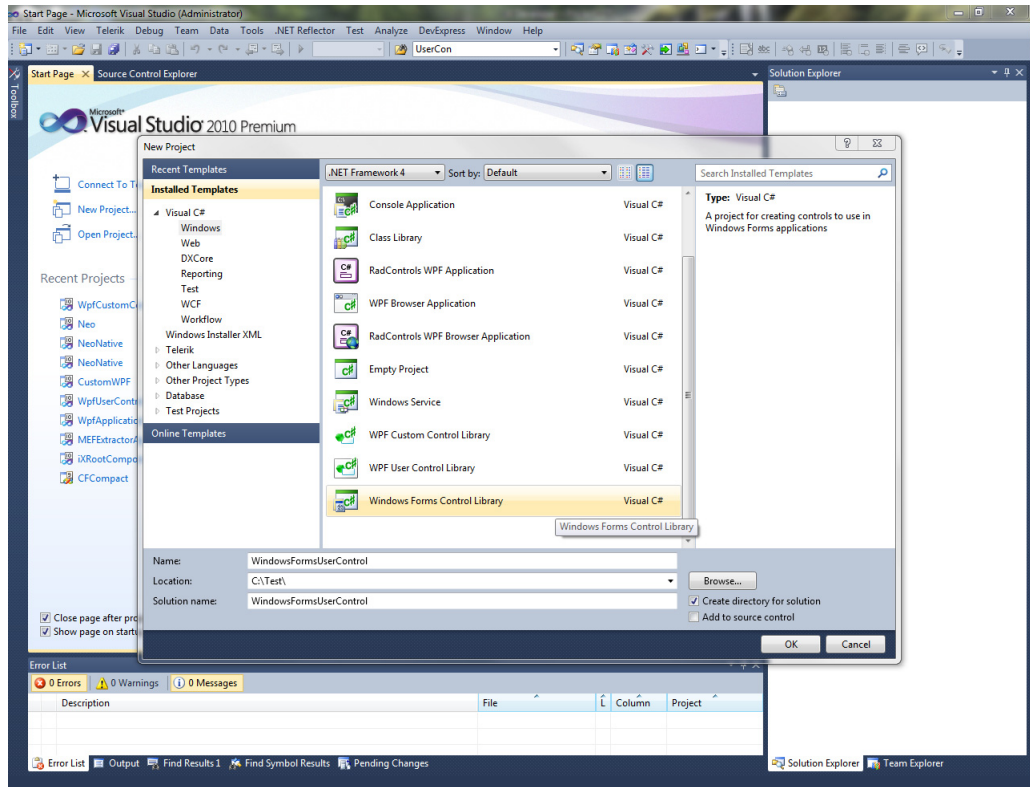
2. Add a TextBox and a Button to the design surface.
3. Add Event Handler for Button click.
4. Add Event Handler for TextBox lost focus.

5. Add a Value Property and INotifyPropertyChanged implementation:

```csharp
public partial class SampleUserControl : UserControl,
INotifyPropertyChanged
{
    public SampleUserControl()
    {
        InitializeComponent();
    }

    public object Value
    {
        get { return textBox1.Text; }
        set
        {
            if (value != null)
            {
                textBox1.Text = value.ToString();
            }
            FirePropertyChanged("Value");
        }
    }

    private void OnButtonClick(object sender, EventArgs e)
    {
        Value = "0";
    }

    private void OnLostFocus(object sender, EventArgs e)
    {
        Value = textBox1.Text;
    }

    public event PropertyChangedEventHandler PropertyChanged;
    public virtual void FirePropertyChanged(string propertyName)
    {
        PropertyChangedEventHandler handler = PropertyChanged;
        if (handler != null)
        {
                handler(null/*this*/, new
                PropertyChangedEventArgs(propertyName));
        }
    }
}
```

6. Use the following code to connect the control to a tag value in iX Developer:

```
public partial class Screen1
{
    void Screen1_Opened(System.Object sender, System.EventArgs e)
    {
        // Hook up value change for a tag
        Globals.Tags.Tag1.ValueChange += OnTagValueChanged;
        // Hook up Property Change on the User Control
        SampleUserControl1.PropertyChanged +=
        OnUserControlValueChanged;
        // Set initial value
        SampleUserControl1.Value = Globals.Tags.Tag1.Value;
    }
        private void OnTagValueChanged(object sender,
        Neo.ApplicationFramework.Interfaces.Events.ValueChangedEventArgs
        e)
    {
        SampleCEUserControl1.Value = e.Value;
    }

        private void OnUserControlValueChanged(object sender,
System.ComponentModel.PropertyChangedEventArgs e)
    {
                Globals.Tags.Tag1.Value = new
                VariantValue(SampleCEUserControl1.Value);
    }


 void Screen1_Closing(System.Object sender,
 System.ComponentModel.CancelEventArgs e)
    {
        // Always remember to unhook the event handlers, otherwise a
        //memory leak is generated
        Globals.Tags.Tag1.ValueChange -= OnTagValueChanged;
        SampleUserControl1.PropertyChanged -=
        OnUserControlValueChanged;
    }
    }
```

The code shows how the value is set on the user control when the tag changes its value, and how the tag value is changed when the user control changes its value.

7.9.5 Creating a Windows Forms User Control for a CE Target

The following example describes how to create a Windows Forms user control designated for a CE target (an operator panel).

1. Start Visual Studio 2005 or 2008 to create a new Smart Device Project.
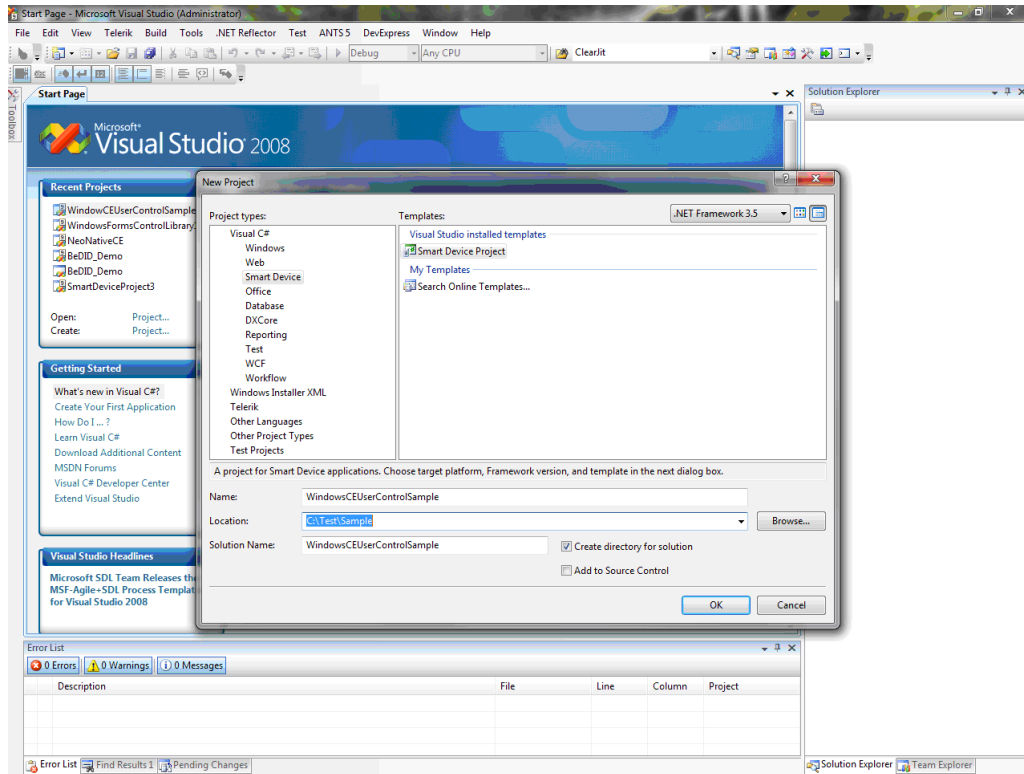
2. Select **Windows CE** for Target platform.

3. Select **Control Library**.
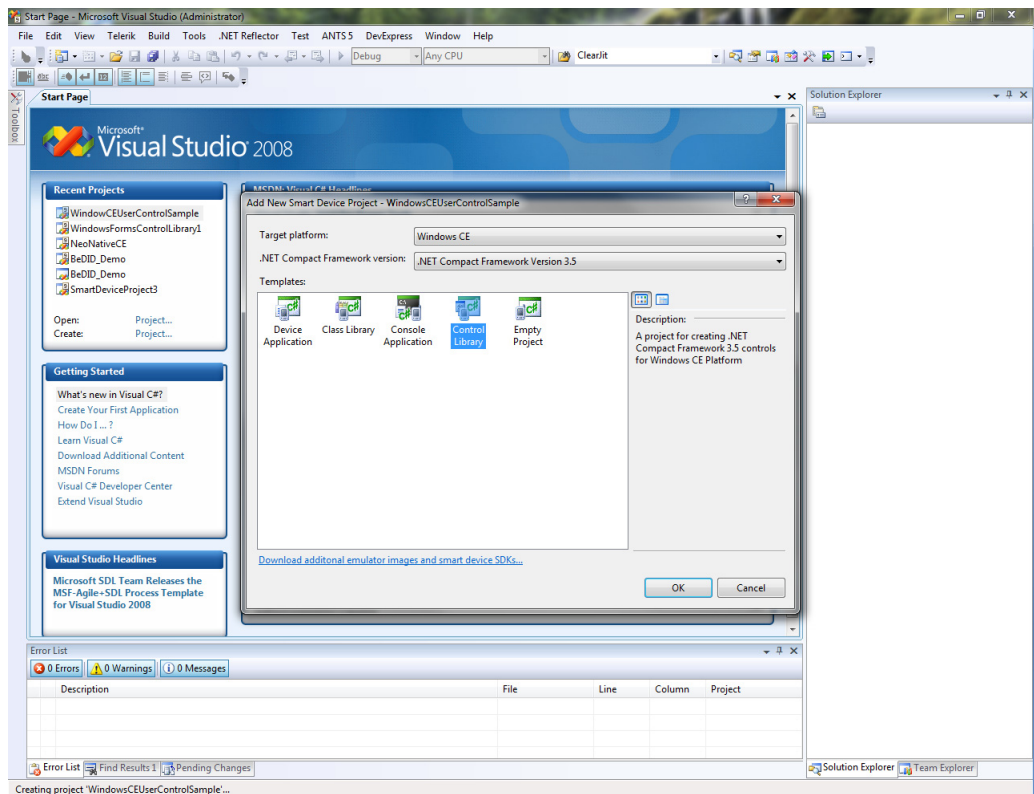
4. Use the same code as in the *Creating a Windows Forms User Control for a PC Target* example.

Note:
Always test your code on the target platform, as properties/methods currently not supported may be included in the code. See *Limitations* for details.

---

8 Ribbon Tabs

Each of the ribbon tabs holds a set of one or more groups. Each group holds a set of one or more controls that are used to manage screens and functions in a project.

The behavior of the ribbon tabs can be configured to be context dependent using the **Options** button in the File menu.

Double-click on any of the ribbon tab headings to collapse the ribbon area. A double-click on a ribbon tab, when the ribbon is collapsed, brings back the original style. This can also be changed from the context menu of the Quick Access toolbar. Tooltips are displayed for all controls on the ribbon tabs. The tooltip offers a brief description of the selected control.

iX Developer contains the following ribbon tabs:

Home Ribbon Tab
Project Ribbon Tab
System Ribbon Tab
Insert Ribbon Tab
View Ribbon Tab
Dynamics Ribbon Tab
General Ribbon Tab
Actions Ribbon Tab

Related information
Options
Quick Access Toolbar

8.1 Home Ribbon Tab

The Home ribbon tab contains groups of controls for editing screens in the project:

Clipboard Group
Screen Group
Objects Group
Font Group
Format Group
Tag / Security Group
Name Group

8.1.1 Clipboard Group

The Clipboard group contains the paste, cut, copy and format painter controls.
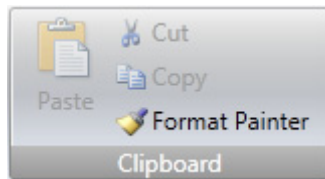
Cut

Cuts the selection to the clipboard. Also available from the keyboard with [**Ctrl**] + **x** or from the context menu.

Copy

Copies the selection to the clipboard. Also available from the keyboard with **Ctrl**] + **c** or from the context menu.

Paste

Copies the clipboard to the current screen. Also available from the keyboard with [**Ctrl**] + **v** or from the context menu.

Note:
Text from a compatible source, e.g. clipboard text, will be converted to a Text Box object when pasted into a screen.

Format Painter

The Format Painter control is used to copy formatting from one object to another. This is useful when creating a uniform appearance for objects.

To apply formatting to another object:

1. Click on an object with the desired format and font properties. Click on the **Format Painter** control. The mouse pointer changes to a paintbrush.

2. Click on the object to format.

The format properties from the first object is applied to the selected object.

Format and font properties can be copied between objects of different types, for example from a rectangle to an HMI control.

Note:
Modification to an object using the **Other Colors** control is not applied when using the Format Painter.

Related information
Other Colors

8.1.2   Screen Group

The Screen control group contains the add screen, background screen, delete screen and current language commands. Screens can also be added and deleted from other controls, such as the Navigation Manager and the context menu of the Project Explorer.
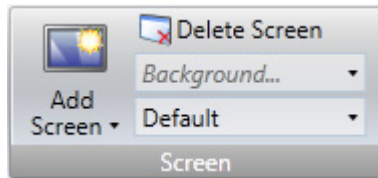
Add Screen

The **Add Screen** command creates a new empty screen that opens for editing. Clicking the lower part of the Add Screen button shows a selection of screen templates are available for selection.

Delete Screen

The **Delete Screen** command removes the current screen from the project, and also disconnects any associations from other screens to the removed screen.

Note:
A deleted screen cannot be restored with the *Undo* command.

Background Screen

Any screen can be used as background to the current screen.

All objects in a background screen will work as designed in the project application. Objects in a background screen must be edited in the screen that they reside in. Objects in a background screen will be dimmed in the current screen in the development environment.

Current Language

Current language is the language that is being used at the particular moment that the project is designed. If multiple languages are not used in the project, only **Default** is available.

Related information
Screens
Language Management

8.1.3 Objects Group

The Objects control group contains static and dynamic objects that can be added to a screen.

Related information
Objects

8.1.4 Font Group

All fonts available in the development PC is available for selection when designing the project. All fonts used in the project will be included when downloading the project to the target.

Note:
The number of fonts needs to be considered when calculating the total size of a project.

Note:
The default font Tahoma does not support Chinese characters in operator panel projects in runtime. The font has to be changed to a font that supports Chinese characters, e.g. MS Song. Keep in mind that some fonts require a lot of project memory.

The **Font** group is used to change text style. The text style is defined for the currently selected object.
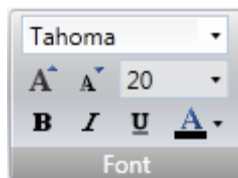
Font and Font Size

Select font from the drop-down list. Use the increase/decrease font size buttons or set a specific font size.

Note:
Objects, for which **Auto Stretch Font** is selected, will not be affected by these settings. The font size will be autosized when resizing the object height.

Style and Color

An italic, underlined or bold style can be selected for the text. The style stays regular if all the boxes are left unchecked. Use the Font Color drop-down list to set the color of the text.

8.1.5   Format Group

The Format group controls the color and position of a selected object. Predefined quick styles offer selection of uniform appearance for different objects. Selection of multiple objects makes it possible to easily affect more than one object at the same time.
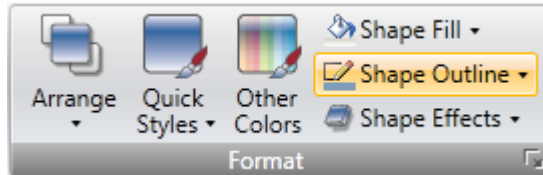
Clicking the small arrow in the lower right corner of the Format group displays the shadow, fill and outline properties with a number of additional format settings.

Note:

It is possible to use the Format Painter (in the Clipboard group) to copy color and font properties from one object and paste it to another object.

Related information
Selecting Multiple Objects

Default Appearance

It is possible to select a style to be used as the default style next time the same kind of object is added.

Right-click on an object to display the context menu, and select **Save Default**, to use the current appearance of the object as the default style. To make existing objects adopt the default style, select **Load Default**. To return to normal mode, select **Clear Default**.
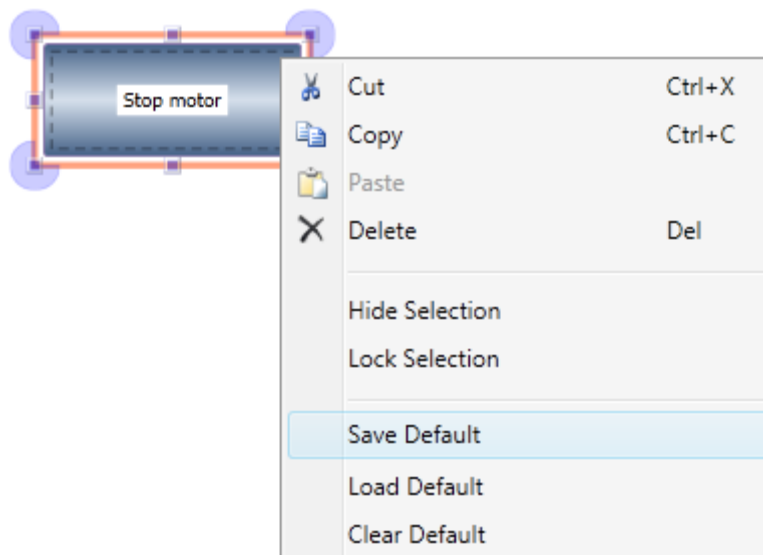
Related information
Format Painter

Arrange

The Arrange control can be used to order, group, resize, align and distribute screen objects.
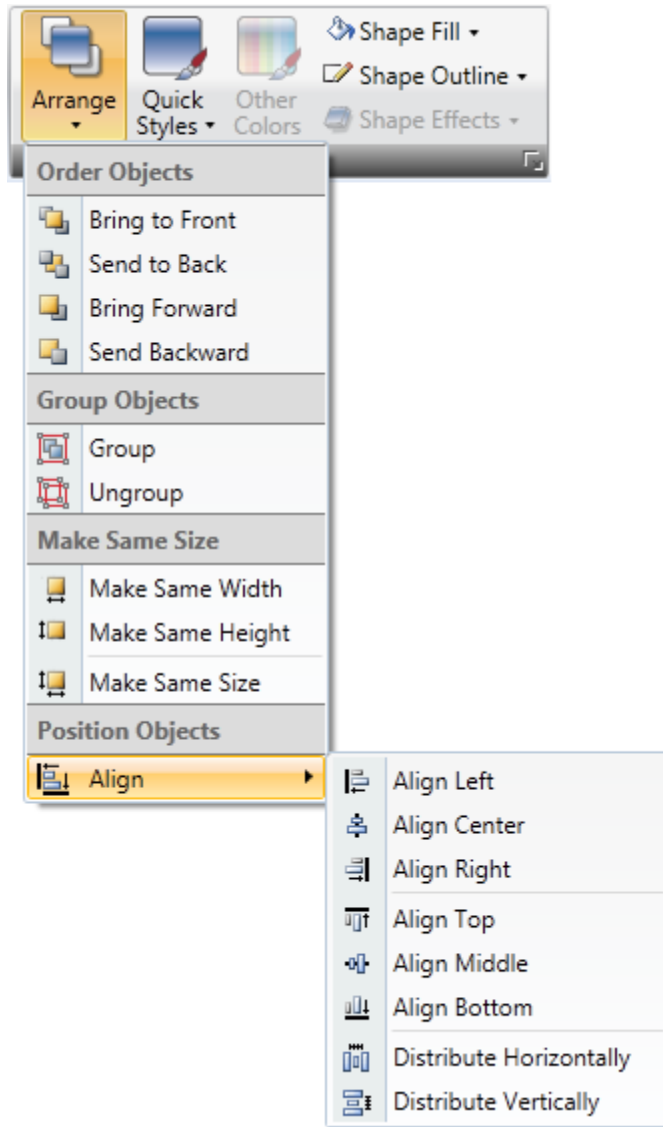
Order Objects

The commands in the Order Objects group are used to send objects to the front or back of each other.

Command	Description
Bring to Front	Positions the selected object on top of any overlapping objects
Send to Back	Positions the selected object behind any overlapping objects
Bring Forward	Makes the selected object swap place with the nearest overlapping object
Send Backward	Makes the selected object swap place with the nearest obscured object

Note:
The Object Browser offers an easy way of arranging objects to the front or back of each other.

Related information
Object Browser

Group Objects

The **Group Objects** commands are used to manage selections of multiple objects. A grouped object is moved and treated as one object with respect to formatting and resizing, and can also be stored in the Component Library for reuse.

Any selection of objects on the screen can be grouped.

When multiple objects are selected, one object is the primary selection. This is shown with an orange frame, while other objects in the group have blue frames. Click on any object in the group to change this to the primary selection.

A grouped object can be resolved into individual objects using the **Ungroup** command. A property that was given to the group will be kept by each of the object, provided that the property is applicable.

Make Same Size

The **Make Same Size** commands are used to adjust the size of a group of selected objects. The size will be determined by the *primary selection* (the guide object).

Position Objects

The **Position Objects** commands can be used to align and distribute multiple objects neatly. Objects can be aligned vertically and horizontally using the primary selection for the group as guide object. Objects can be aligned with respect to their center (center/ middle) or their edges (top/bottom/right/left).

Related information
Component Library
Snap Objects to Other Objects
Grid

Quick Styles

The **Quick Styles** control contains a number of preset visual object color schemes that can be applied to screen objects.

The predefined graphical styles make it possible to achieve a uniform appearance on items like meters and trend viewers. An object that has a style applied can be further customized, for example with font settings.

It is also possible to customize the predefined styles to styles of your own preference. The quick styles are not specific to the current project. They will be available for all iX Developer projects.

Note:
Controls in the Style group on the General tab can be used to reshape objects such as meter, slider and trend viewer to predefined styles. These styles are not the same as the quick styles.

Related information
Font Group

Other Colors

Color settings for special properties of an object, for example scale color for a meter or out-of-range validation of an analog numeric object, are available from the **Other Colors** control.

Shape Fill

The **Shape Fill** control is used to alter the fill color and gradient of the selected shape. A preview of the colors in the Shape Fill selection window is displayed directly on the object on the screen as the pointer hovers over the different colors.

Shape Outline

The **Shape Outline** control is used to alter the outline color and outline width of the selected shape.

Shape Effects

Shape Effects makes it possible to add bevel style and shadow effects.

Note:
Bevel in combination with fill color "no fill" is only supported for Panel TxC and PC targets. The bevel will not be visible in runtime on other operator panels targets.

8.1.6   Tag/Security Group

Connection to controller tags as well as object access (based on security groups) and visibility are configured from the **Tag/Security** group.
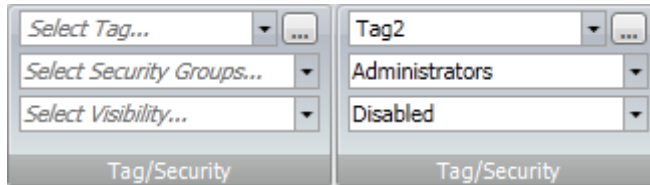
Tag/Security group without and with selections

Parameter	Description
Select Tag	Connection to a tag from the tags configuration page. Tags may also be added directly, but will be remain internal until connected to a controller tag. Click **...** to configure expressions.
Select Security Groups	Allows limiting access of the object based on security groups. If left empty, the object can be accessed by anyone.
Select Visibility	Configuration of visibility: **Default**: The visibility specified on the General tab of the Security Manager is used **Disabled**: The object is disabled for users with insufficient access rights **Hidden**: The object is invisible for users with insufficient access rights **Normal**: The object looks normal also for users with insufficient access rights

Note:
When access rights are not fulfilled, showing an access denied message or opening a login dialog can be configured using the security function.

Related information
Expressions
Adding Tags during Editing
Security Management

8.1.7   Name Group

All objects and screens are automatically named when they are created. The name is displayed and can be edited in the Name group. Screen names can also be edited in the Project Explorer.

The name of an object is a symbolic identifier and is an alphanumeric string, beginning with a letter. Name strings can contain letters (a-z, A-Z), numbers (0-9), and underscore ('_') characters.
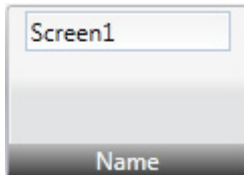
Note:
Renaming objects and screens included in scripts will cause the scripts not to function.

Related information
Project Explorer
Invalid Names

8.2 Project Ribbon Tab

The Project ribbon tab contains a number of project related functions, divided into the following groups:

Run Group
Transfer Group
Project Group

8.2.1 Run Group

The Run group enables validation, rebuilding, simulation and debugging of the project.
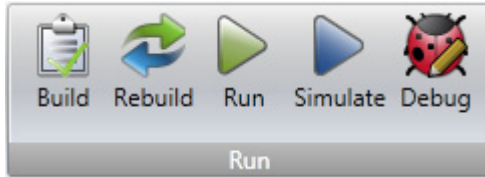
Build

The build control validates the script code and compiles the current project.

Recompiling the project using the Build control will result in only the changes being made to project to be recompiled. To fully recompile the project, instead use the Rebuild control.

Rebuild

The rebuild control validates the script code and recompiles the complete project.

Run

A test of the project can be executed directly on the PC. Clicking **Run** will perform a validation and start a simulation, if the project can be built. If errors are found during validation, an error message will be displayed.

During simulation, the project will communicate directly with the configured controllers, provided that the controllers are correctly connected to the development environment.

Note:
If project build fails, check that the absolute path to the project, including its components, does not exceed 260 characters.

Simulate

It is also possible to test and run the application on the development PC without connection to the selected controller, by clicking **Simulate**. If the project is valid, the simulation is started.

Debug

The Debug command will allow the user to debug scripts in a real c# debugger with the possibility to set breakpoints etc. The command builds and checks the current project, and launches the application associated with the generated file. The function scans the development PC for an installed debugger. If no debugger is found, a link for downloading debugger is presented. User preferences for the script debugger may be entered under **Options**, available from the File menu.

Note:

Visual Studio 2008 requires SP1 in order for associated files to be opened automatically (i.e. using the Debug command).

Debugging of a project with Script module included is not supported using Visual Studio 2010.

Debugging of a CE project is not supported using Visual Studio 2010.

Related information
Output
Database
Options

8.2.2 Transfer Group

The Transfer group enables download to operator panel, upload database and export to folder functions.
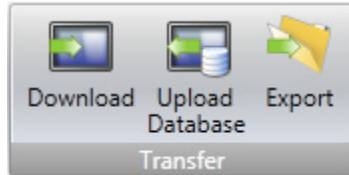
Note:
Time zone, region or daylight saving settings will not be transferred if these settings have been changed manually in the panel.

Download

The Download control sends the project to an operator panel or a PC with an iX
Runtime Installation, in the network environment.

Parameter	Description
Connected targets	Choose a target type to transfer the project to.
Download	Start the project transfer.
	Click the **"Download all source code (for later upload) "** checkbox to compress the project and save it as a ZIP file on target.
	Click the **"Password protected"** checkbox and enter a password to protect the ZIP file.
	Download is supported for Panel TxA, TxB, TxC and TA targets.
Verify	The verify function enables the user to check if a target platform configuration is identical to the project configuration.
	Verify is supported for Panel TxA, TxB, TxC targets. If greyed out, please update to latest "System program".
Log	Show the data being logged during the download and verify process.

If the target for the download contains newer files or files that don't exist in the project, a dialog will appear prompting for which files to be kept, overwritten or deleted.
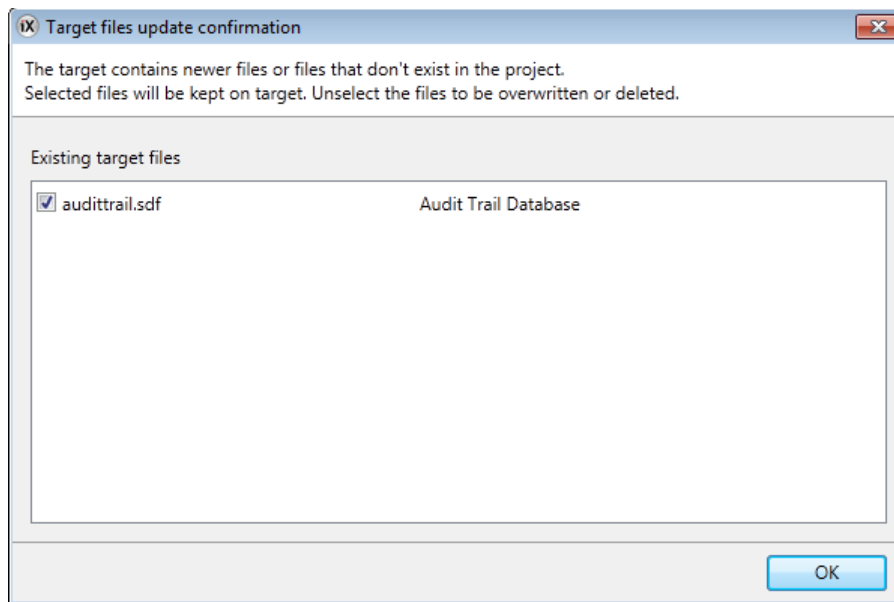
Note:

If the same project is to be downloaded to a panel again, the original datalogger in the panel will be replaced if any settings have been changed in the datalogger since the last download.

Downloading a Project to an Operator Panel or a PC with an iX Runtime installation

1. Connect the operator panel to the PC via an Ethernet connection.
2. Click **Download**.
3. The download dialog opens and all panels connected in the network are listed with information about IP address and panel type. Select the panel to download to in the list.
4. Check that the IP address is correct and click **Download**.
   a. Optional: Click the "**Include compressed project**" checkbox to compress the project and save it as a ZIP file.
   b. Optional: Click the "**Password protected**" checkbox and enter a password to protect the ZIP file.

The panel will respond to a download request by stopping any current project. The panel will display that it is ready for project download. When the download is completed, the new project will automatically start running.

The project folder is located at the following path. Only for PC targets!

%public%\Documents\Beijer Electronics AB\iX Developer Runtime\Project

---

Note:
The panel settings can be controlled from the panel's Service menu.

Related information
Service Menu

Upload Database

The Upload Database command will collect the database from an operator panel in the network environment to a file on the development PC. The panel will pause during the upload, and will be started automatically afterwards.

Note:
The database may only be collected from a panel running the same project (i.e. a project with the same name and of the same size) as the current iX Developer project.

Collecting a Database from an Operator Panel

1. Connect the operator panel via an Ethernet connection.
2. Click **Upload Database**.
3. The Upload Database dialog opens and all panels connected in the network are listed with information about IP address and panel type. Select the panel to collect from in the list.
4. Check that the IP address is correct and click **Backup**.
5. Select location of the saved database.

Export

Exports the project to a folder. The export dialog allows creating a new folder.

Exporting a Project to an Operator panel or a PC with an iX Runtime Installation

A project can be exported to an Operator panel or a PC with an iX Runtime installation via the **Export** command. The project can be exported to an USB stick or, on the condition that a network connection is configured, directly to a folder in the runtime PC.

The project in the runtime PC is started by running the .exe file located in the project folder.

There will also be created a **TransferClient** .exe file, a binaries folder and a configuration file. The **TransferClient** is a standalone application for downloading the exported project to a target. It works in the same way as the **Download** control from inside iX Developer. It is possible to view a log for the export by clicking the **Log** button on the **TransferClient** dialog.

Related information
Download

8.2.3   Project Group

The project target as well as name and title are available for editing in the Project Group. The project title is displayed in the title bar in runtime by default. Leaving the title empty displays the screen name/screen title only in runtime.

The Project Group also contains the Compress Project command, a Project Settings control and the possibility to refer to other assemblies using the Referenced Assemblies control.
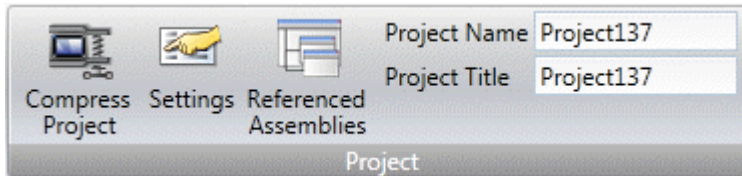
Related information
Compress Project
Settings
Referenced Assemblies

Compress Project

The project source files can be compressed into a Zip file using the **Compress Project** command. The Zip file can be saved to an USB stick or, on the condition that a network connection is configured, directly to a folder in the runtime PC
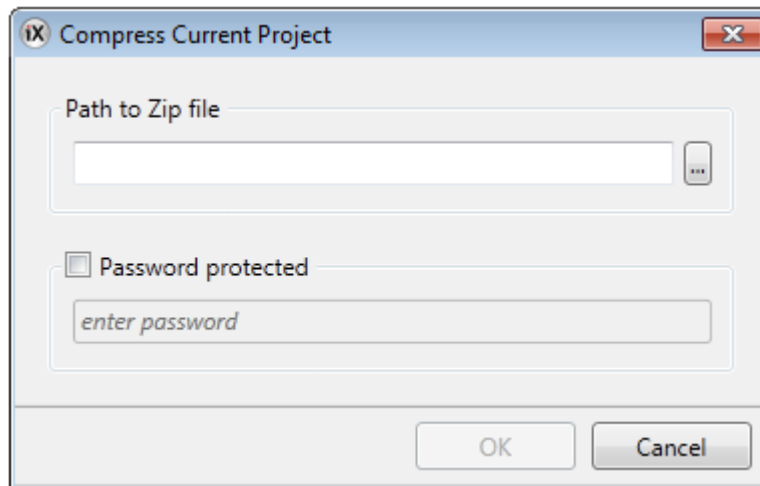
Parameter	Description
Path to Zip file	Browse to the location to save the Zip file.
Password protected	Optional password protection of the Zip file.

Related information
Opening a Project

Settings

Properties such as application behavior in runtime, system font and settings for mouse and keyboard can be edited by clicking **Settings** in the Project group.

Related information
Alarm Button
Deployment
Display Panel
Mouse / Touch Input Delay
System Font
Virtual Keyboard
Screen Name and Screen Title

Alarm Button
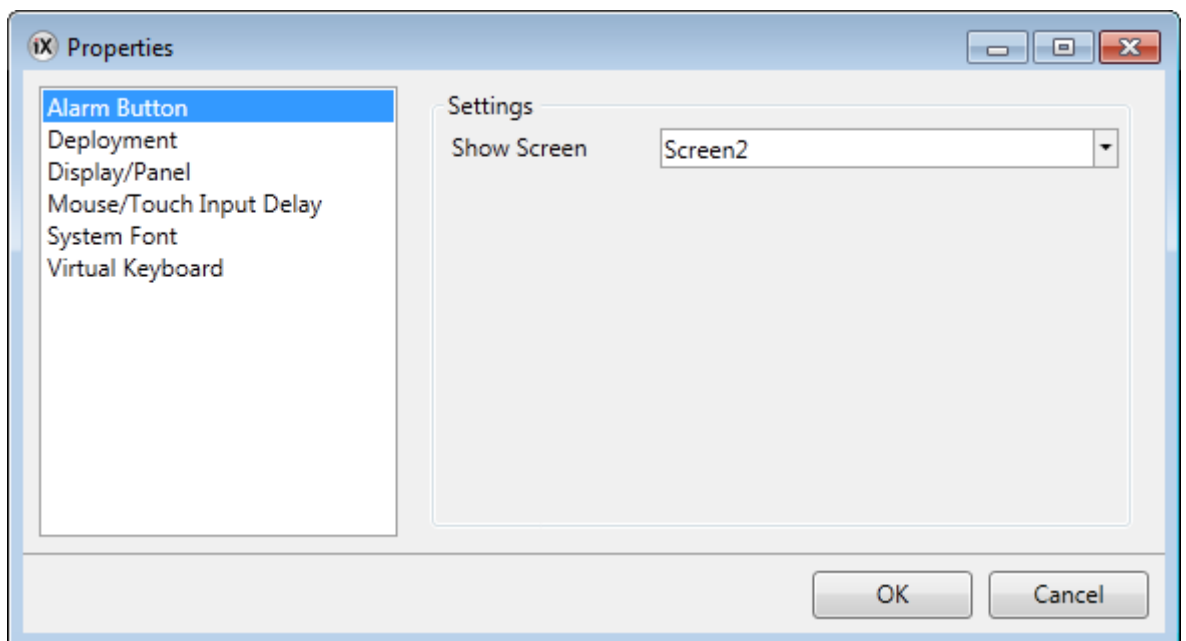
Parameter	Description
Show Screen	Selection of screen to be displayed when the predefined Alarm key on a keyboard operated panel is pressed.

Deployment
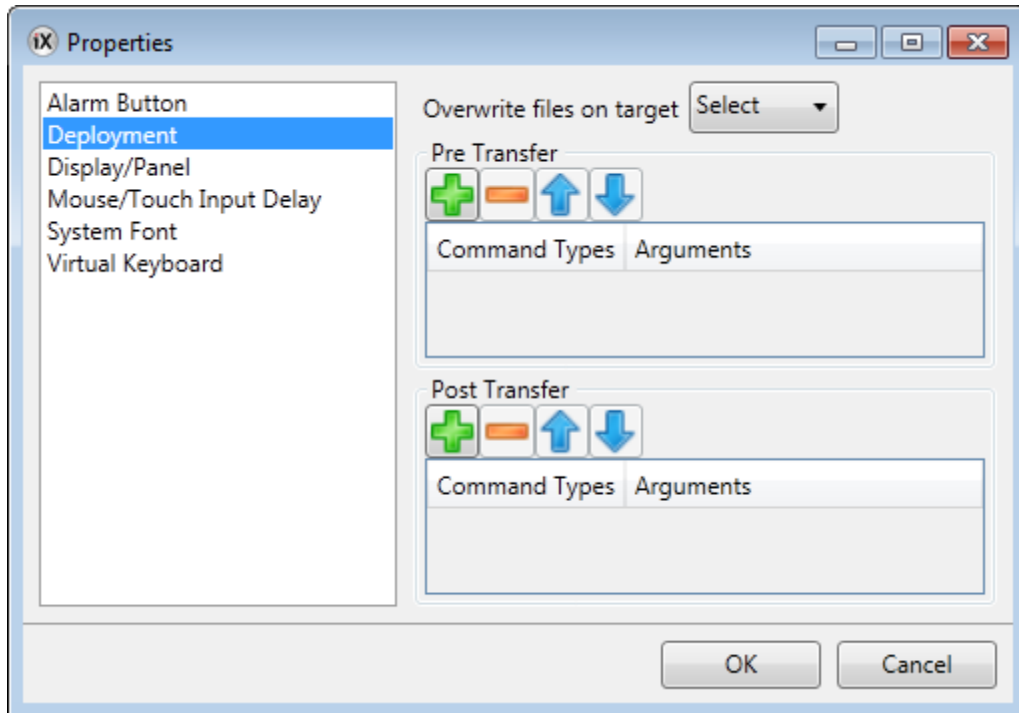
Parameter	Description
Overwrite Files on Target	This setting can be used to determine the default behavior when transferring projects and the transfer finds critical files already present on the target.
Pre Transfer	Add commands to be executed before the file transfer. Only available for standard PC targets.
Post Transfer	Add commands to be executed after the file transfer. Only available for standard PC targets.

Related information
Pre Transfer
Post Transfer

Pre Transfer

The following commands can be executed before the file transfer:

Command	Description
Reboot	Reboots the target. Possible arguments: **True** and **False**. Using **True**, the target tries to start any project present. The argument **False**, or no argument, prevents the target from starting the project automatically.
Wait	Pauses the execution of the commands. This can be useful if the user runs commands that takes time for the target to recover from. Possible arguments: a number, indicating how many whole seconds to wait.
Run	Starts any given process. The first argument must always be the path to the executable to be run. Keep it inside quotes if it contains spaces. Other arguments: **/p:** or **-p:** are parameters for the process to start. The parameters must be inside quotes and separated by spaces. **/noshell** or **-noshell** means that the process should be run outside of the command shell. **/wd:** or **-wd:** means working directory for the process to run. If left empty will default to active projects directory. The parameters must be inside quotes. **/wait:** or **-wait:** means that the **TargetControlService** should wait until the process has exited before continuing. If the process does not exit by itself, this argument should be left out.

It is possible to use the following **macros** in conjunction with the **Run** commands:

Macro	Description
$projectpath$	Replaced by target project folder. If the user has downloaded an executable together with the project and wants to run it, this macro represents the destination folder. Example: **"$projectpath$\dbbackup.exe" /p:"backup.bak" -noshell -wait** This will run the **dbbackup.exe** with the startup argument **backup.bak**, it will run outside the **command shell** due to **-noshell**, and the transfer will not continue until the process has exited due to **-wait**.
$startproject$	Registers and starts the project. The transfer will continue.
$finished$	Tells the system that the transfer is finished and closes connections. The project will not start. Should only be used in Post Transfer.

Note:
Only available for Panel TxC and PC targets.

Post Transfer

The same commands and macros as for the Pre Transfer can be used, but are executed after the file transfer.

Note:
Only available for PC targets.

Related information
Pre Transfer

Display Panel
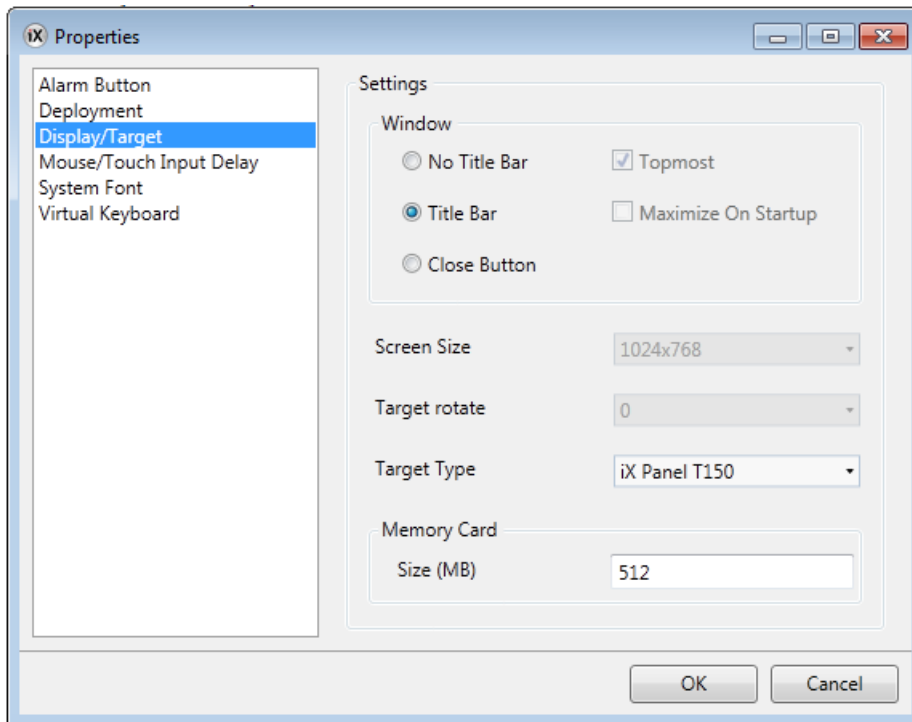
Parameter	Description
No Title Bar	Displays the project without title bar in runtime.
Title Bar	Displays the project with its title and a set of minimize, maximize and close buttons in the title bar.
Close Button	Displays the project with its title and a close button in the title bar.
Topmost	Makes the project the topmost application in runtime. Only available for Panel TxC and PC targets.
Maximize On Startup	Makes the project maximized in runtime. Only available for PC targets.
Screen Size	Screen resolution. Only available for standard PC targets.
Panel Rotate	Displays selected rotation. It is possible to change the rotation of the panel and panel content by 0, 90, 180 or 270 degrees. Only available for Panel T7A, T10A and TxC targets.
Panel Type	Displays selected target. It is possible to select another target.
Memory Card	State the size of the internal memory card, to save project data on it. Only available for Panel TA70 - TA150 targets.

Related information
Changing Project Target
Memory Card

Changing Project Target

When changing to another target, you can select to automatically resize objects to the new target size. Different scale factors are used horizontally and vertically, to optimize conversion to and from wide screen targets. In this case, the aspect ratio of objects will change.

The following items are resized with the same factor when having selected automatic resizing:

Item
Graphical objects
Popup screens
Font A minimum scaled font of 10 pixels ensures readability in the smaller operator panels
Dynamics size and move

Not all objects and styles are supported for all project targets. When changing to another project target, objects and styles that are not supported in the new target will be deleted and will not possible to restore. Removed objects are logged, and displayed in a window after the transformation. The log may be saved to file. Build files are always cleaned and project saved after transformation.

Note:
Changing back to the previous target will not restore removed objects.

Memory Card

The available project memory is displayed in the lower right corner of the application window, when the target is an operator panel. The project memory size is updated when the project is validated.

Project size: 8,1 / 522,0 MB Tags used: 10

If the size of the memory card is not stated, project data will be saved in the internal operator panel memory.

For panels with a pre-installed internal memory card, the size of the memory card is added automatically.

Mouse/Touch Input Delay

It is possible to configure a delay of mouse and touch screen input in runtime. This setting is global and will affect all objects for which **Delay Mouse/Touch Input** has been selected. The Property grid is used to enable delay of mouse and touch screen input for individual objects.
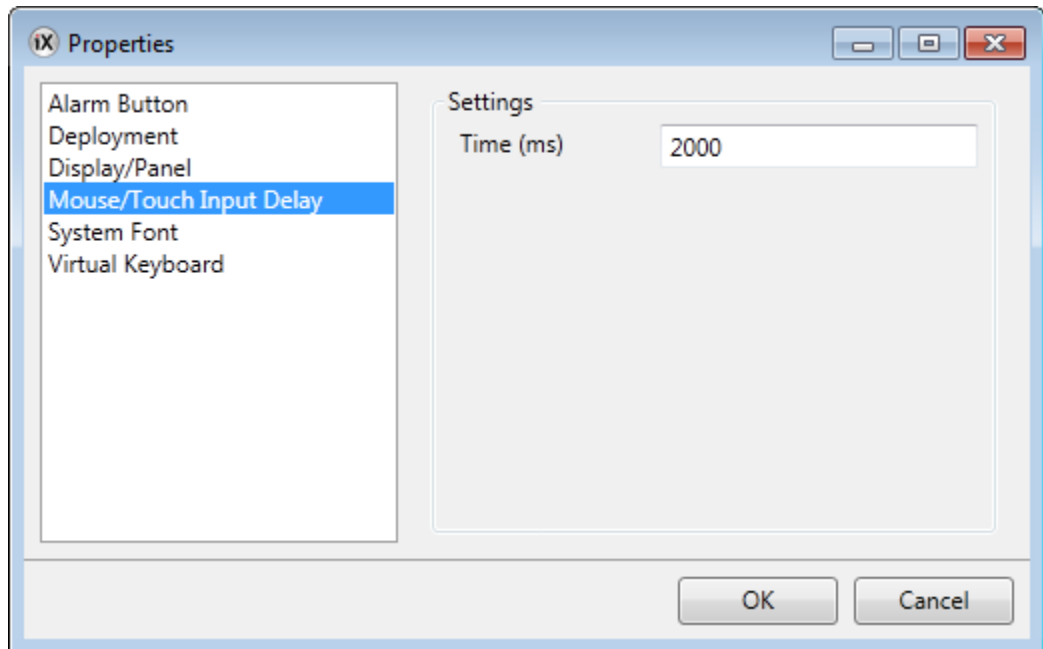
Related information
Property Grid

System Font
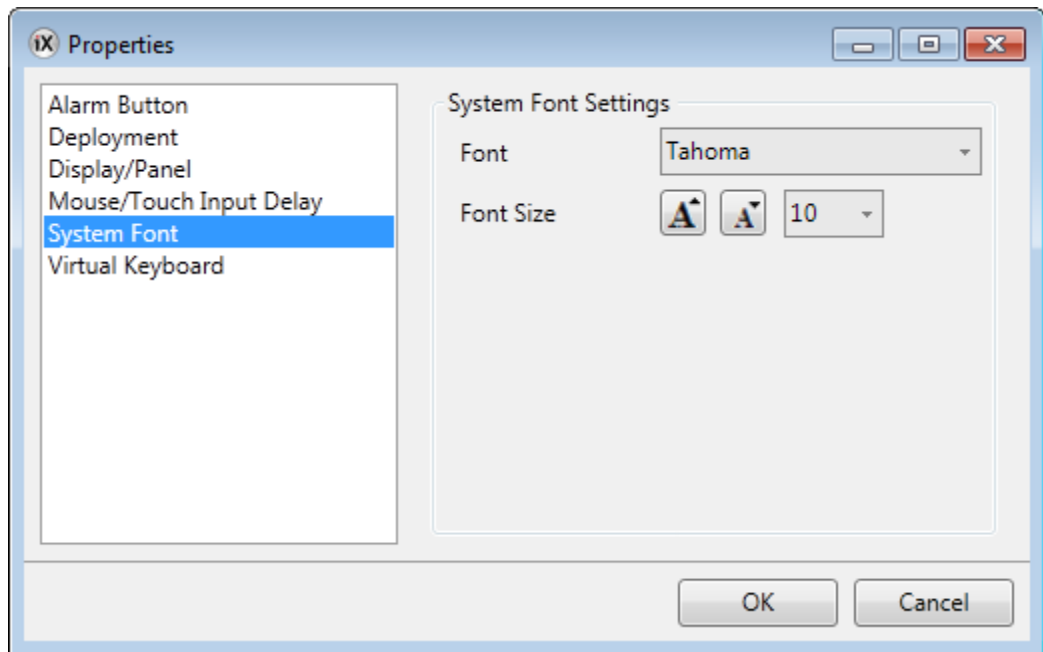
Parameter	Description
Font	Select a system font that will be used as the default font.
Font Size	Use the increase/decrease font size buttons or set a specific default font size.

Virtual Keyboard
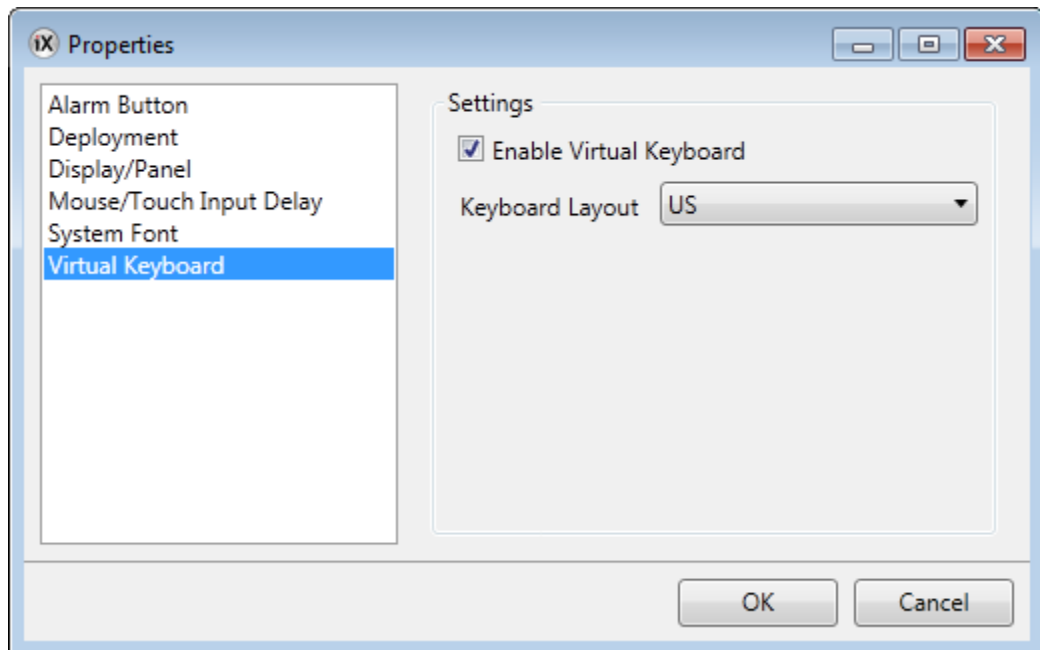
Parameter	Description
Enable Virtual Keyboard	Select whether a virtual keyboard is to be displayed when alphanumerical input is needed.
Keyboard Layout	The layout of the virtual keyboard.

The virtual keyboard that is displayed in runtime depends on settings made for the Multiple Languages function. If only the default language is used, the project settings (above) are used.

Related information
Language Management

Referenced Assemblies

The Referenced Assemblies command makes it possible to refer to your own or third-party .Net assemblies (.dll). Their methods and properties will be made available when scripting and will be included by the name completion function, in the same way as for built-in components.

1. Click on **Referenced Assemblies**.
2. Click **Add**.
3. Browse to the desired dll file in your PC environment and click **Open**.
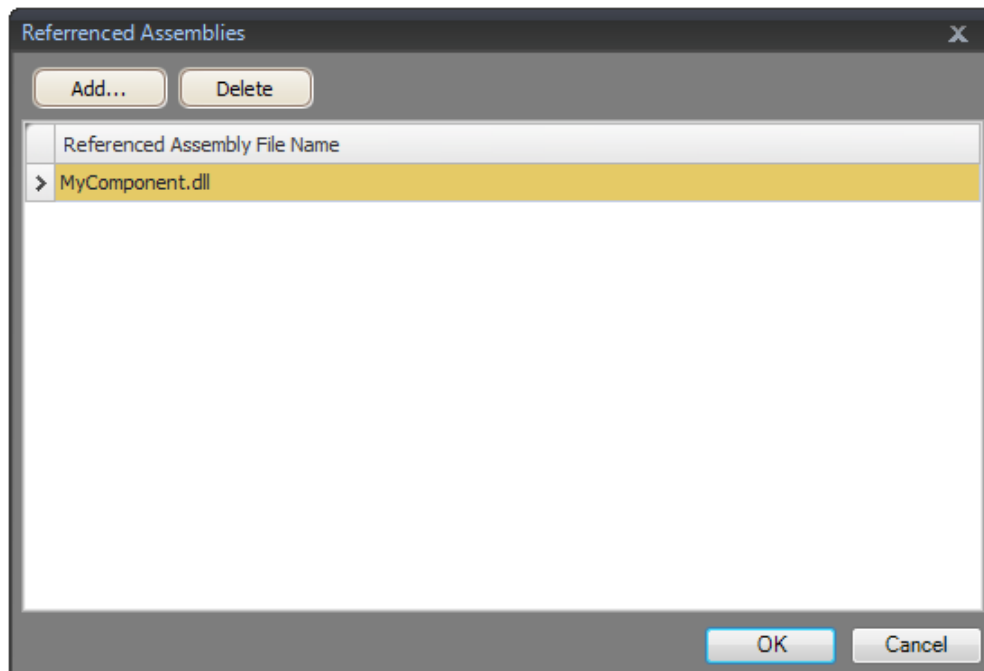
The methods and properties of the added dll file are now available for scripting.

8.3 System Ribbon Tab

The System ribbon tab controls system settings. These settings are divided into the following groups:

Time Zone and Region Group
Buzzer Group
Backlight Group
Serial Ports Group
Servers Group
Output Devices Group
Service Menu Group

8.3.1   Time Zone and Region Group

National settings are made in the Time Zone and Region group.
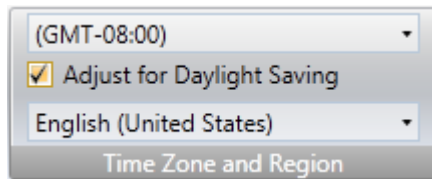
Select time zone, whether to adjust for daylight saving, and region. If no time zone is selected, the time zone that has been selected for the panel or PC will be retained.

The date and time display format (for example using AM/PM for time) is based on the operating system settings. For operator panel projects, this format may be altered by selecting another region.

8.3.2   Buzzer Group
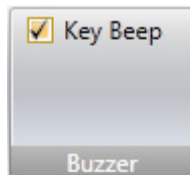
Select whether to let a key beep sound when making keyboard input in runtime. Only available when an operator panel is selected as target.

8.3.3   Backlight Group
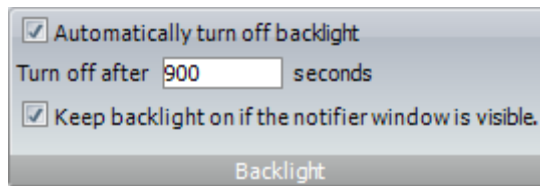
Select to automatically turn off the backlight after a certain number of seconds, and whether to keep the backlight on if a notifier window (e.g. alarm indicator or communication error message) is visible.

Backlight settings are supported only for operator panel targets. For other targets, the Microsoft Windows screen saver may be used.

8.3.4   Serial Ports Group
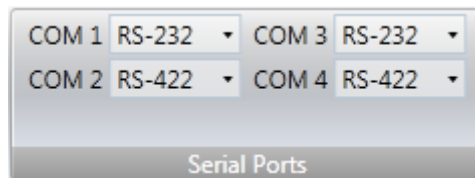
For each of the COM ports, select if it is to be configured as an RS232, RS422 or RS485 port. Only available when an operator panel is selected as target.

Note:
The number of available ports and possible configuration differ depending on selected target.

For Panel TxA, TxB, TxC and IPC, four COM ports can be configured:

Port	Description	Label on back of panel
COM1	RS232 incl. CTS	Port 1
COM2	RS422 or RS485	
COM3	RS232 excl. CTS	Port 2
COM4	RS422 incl. RTS or RS485	

8.3.5   Servers Group

The Servers group contains the FTP, Web Server, OPC UA Server and Remote Access commands. The FTP and Remote Access commands are only available when an operator panel is selected as target.
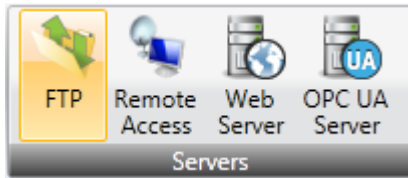
FTP

When enabled, it is possible to upload/download files to/from the operator panel provided that there is an FTP client program in the PC, e.g. Internet Explorer, Windows Commander or some other standard FTP program.

It is possible to set up access to the FTP server with login requirements and/or as anonymous. An anonymous user has read-only access rights. A user that logs in with the correct user name and password gets read/write access rights.

Only available when an operator panel is selected as target.

FTP is not supported for Panel TxC.

Remote Access

The Remote Access function makes it possible to access, reflect and control an operator panel from a PC by using the free VNC client program Remote Access Viewer together with the built-in VNC server in the operator panel.

It is possible to set up Remote Access with a view only password and/or a full access password. The view only password allows reflecting the panel remotely; the full access password allows also control of maneuverable objects in the panel.

Only one VNC client can connect to the VNC server.

Only available when an operator panel is selected as target.

Note:

To protect secrecy when entering a password via Remote Access Viewer, it is recommended to use the PC keyboard. Otherwise it is possible that the cursor on the remote operator panel displays which keys are pressed on the alphanumeric keyboard.

For more information about Remote Access, refer to the documentation for Remote Access and Remote Access Viewer.

Web Server

Enables the possibility to host project files for a web client.
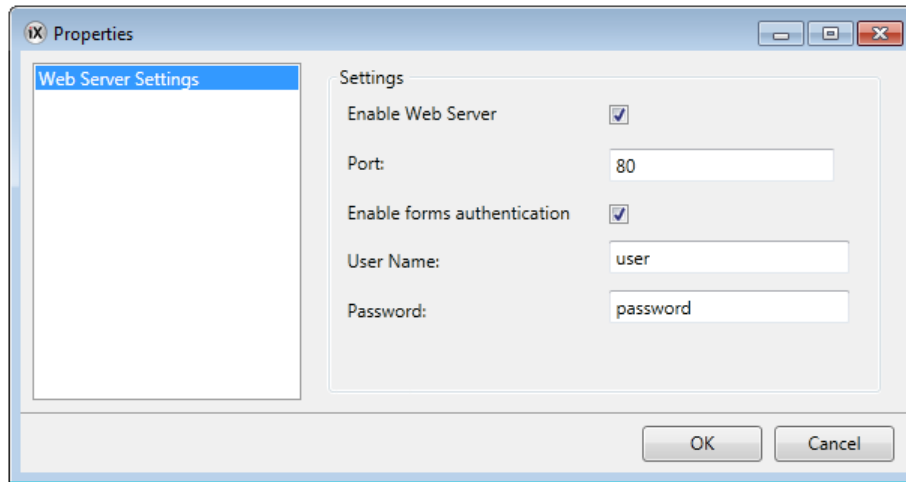
Parameter	Description
Enable Web Server	Click this checkbox to enable the web server.
Port	Enter a server port number.
Enable forms authentication	Click this checkbox to enable forms authentication and enter user name and password. The password can be any alphanumeric string. The passwords minimum length is four characters and the maximum length is 20 characters.

Related information
Web Server

OPC UA Server

Enables the possibility to publish tags with an OPC UA Server.
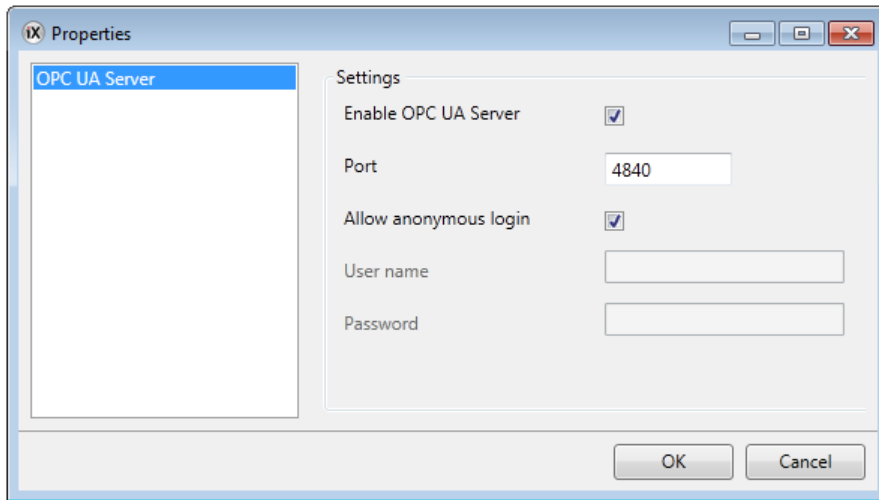
Parameter	Description
Enable OPC UA Server	Click this checkbox to enable the OPC UA server.
Port	Enter a server port number.
Allow anonymous login	Click this checkbox to enable anonymous login. For a secured login enter user name and password and leave the checkbox unchecked.

Address space

All tags, except array tags, defined in project will be shown.
The tags will be put in a folder named Tags in Namespace TagProvider on the server.
The NodeClass of all tags will be Variable.
All tag values will be published scaled with the offset and gain of the tag.

Tag Properties Mapping

Parameter	Description
Tag property	UA Server Attribute
Name	DisplayName / BrowseName
Data Type	DataType
Access Right	AccessLevel / UserAccessLevel
Description	Description
Poll Group Interval	MinimumSamplingInterval

Certificate

A certificate will be auto generated and self-signed when the server first starts up.

The certificate will be valid for 20 years.

The certificate contains no IP address since this could change after the certificate has been generated.

Limitations

Limitation	Description
Discovery	Not supported
Encoding / Transport	Only UA TCP Binary
User authentication	Only user name / password Only one account
Security	No encryption support

Related information
OPC UA Client

8.3.6 Output Devices Group

Printer devices can be set up by clicking the Settings control in the Output Devices group.

Printer Device

USB Printer Connection
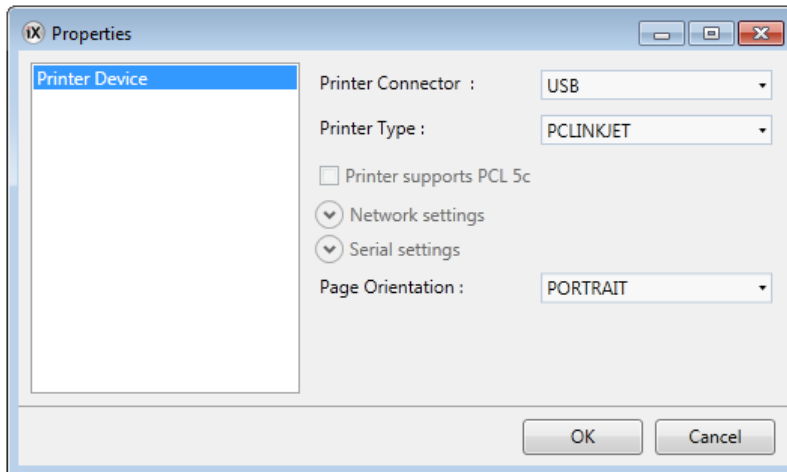
Parameter	Description
Printer Connector	Select USB
Printer Type	Select printer type. Selection possibilities: PCLINKJET and PCLLASERJET.
Printer supports PCL 5c	Click this checkbox if the printer supports the PCL 5c color printer protocol. This option is only available if the printer type is set to PCLLASERJET.
Network settings	Not available
Serial settings	Not available
Page Orientation	Portrait or landscape orientation of the printout

Ethernet Printer Connection
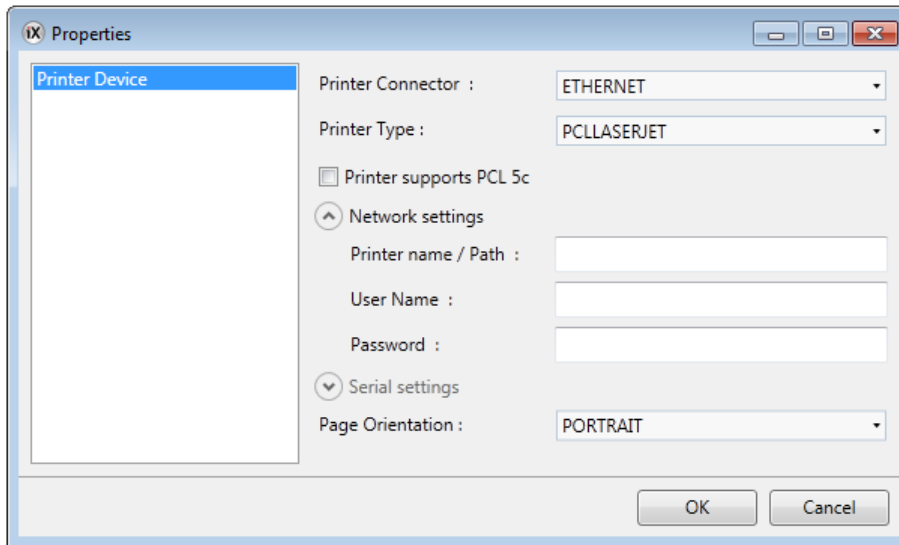
Parameter	Description
Printer Connector	Select Ethernet
Printer Type	Select printer type. Selection possibilities: PCLINKJET and PCLLASERJET.
Printer supports PCL 5c	Click this checkbox if the printer supports the PCL 5c color printer protocol. This option is only available if the printer type is set to PCLLASERJET.
Network settings	Enter Printer name / path
	Enter user name
	Enter password
Serial settings	Not available
Page Orientation	Portrait or landscape orientation of the printout

Serial Printer Connection
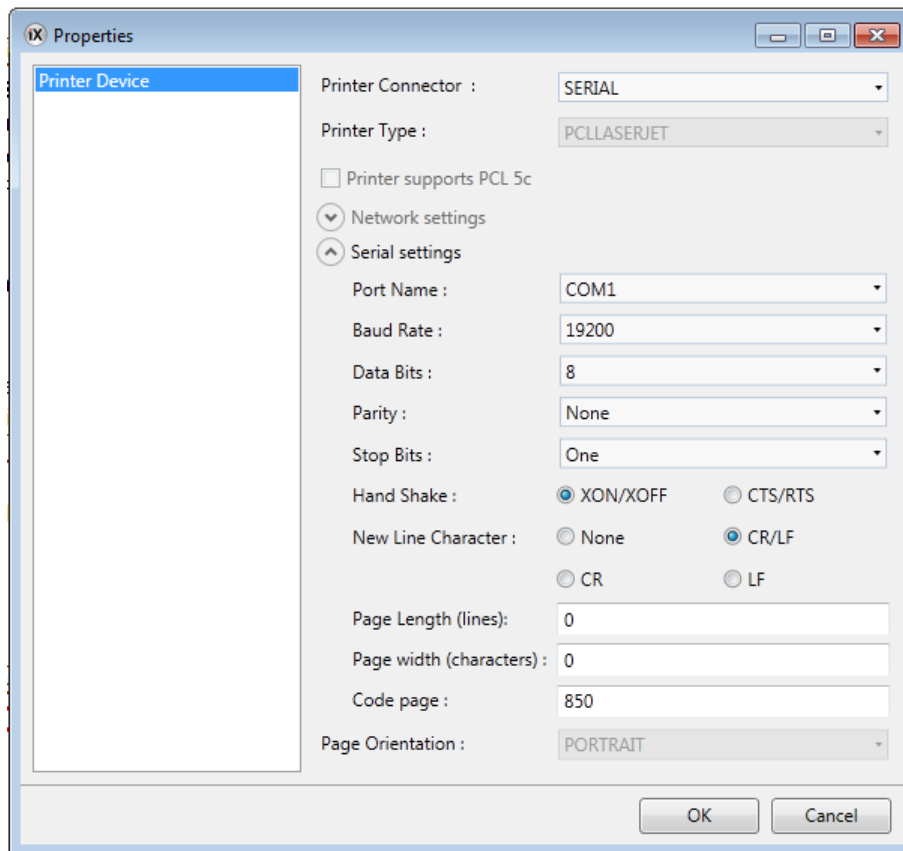
Parameter	Description
Printer Connector	Select Serial
Printer Type	Not available
Printer supports PCL 5c	Not available
Network settings	Not available
Serial Settings	**Port Name:** Select which COM-port the printer is connected to
	Baud Rate: Select the baud rate for transmission of data.
	Data Bits: Select the number of data bits in each character.
	Parity: Set the parity bit. The parity bit in each character can be set to none , odd, even, mark or space.
	Stop Bits: Select the number of stop bits to be sent at the end of every character. Selection possibilities: None, One, Two or TwoPointFive.
	Hand Shake: Select which handshaking protocol to be used. Selection possibilities: XON/XOFF, CTS/RTS.
	New Line Character: Select end of line character. Selection possibilities: None, CR/LF, CR or LF.

Parameter	Description
Serial Settings	**Page Length (lines):** Select the number of lines to be printed out before form feed. Form feed will not take place if the page length is set to 0.
	Page Width (characters): Set the page width specified as the maximum characters in a single line of text.
	Code page: The code page consists of a table of values that describes the character set for a particular language. The default code page in iX Developer is 850.
Page Orientation	Not available

Printouts via Operator Panel

When printing to a serial printer from an operator panel, the printer must support IBM character set (850).

When printing to a USB printer from an operator panel, the printer must support the printer classes according to the table below:

Printer type	Printer class requirement	Recommended printer model
Color laser printer	USB + PCL 5c	HP LaserJet 2700
Monochrome laser printer	USB + PCL 5e	HP LaserJet 1320
Color ink-jet printer	USB + PCL 3e	HP Deskjet 5652

When printing via Ethernet from an operator panel, the network printer must be a shared resource in the Windows network.

Note:
Connecting a printer via Ethernet requires that a PC is connected between the operator panel and the printer.

Note:
Printing to a color laser printer via Ethernet is not supported for operator panels.

Note:
The printer class PCL 6 is not supported at all for connection to operator panels.

8.3.7   Service Menu Group

The Service Menu group provides a possibility to protect the service menu with a pin code.
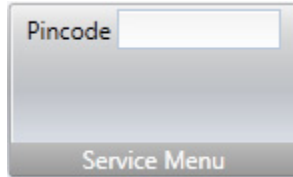
Parameter	Description
Pincode	Enter a pincode for the service menu. No pincode enables the service menu for all users. Only numbers (0–9) are allowed.

Related information
Service Menu

8.4   Insert Ribbon Tab

The Functions group, available from the Insert ribbon tab, contains functions and screen that can be added to the project.

Functions Group

8.4.1 Functions Group

All added components are available from Project Explorer folders. The Project Explorer already includes alarm server, multiple languages, security functions and tags. All functions are managed through their configuration pages.

Parameter	Description
Text Library	With the text library function, text tables can be created, where values are linked to texts.
Data Logger	Data can be logged and saved to a database on time intervals or depending on changed values.
Alarm Distributor	Alarms can be distributed between operator panels, with notification via printer, SMS or e-mail using the alarm distributor.
Script	A script module can be included to program features not covered by included functions and actions, or to be used to share functionality among functions.
Recipe Management	Several recipes can be inserted in the project. Each recipe function handles a pre-defined set of recipe items.
Function Keys	Function keys for operator panels as well as for the regular PC keyboard can be configured.
Scheduler	A scheduler can be used to control events in the process at special times.
Audit Trail	The Audit Trail function enables tracking of operator actions.
Reports	The reports function allows adding Excel report templates to the project.
Screens	The Screen control adds new screens to the current project.

Related information
Project Explorer
Configuration Pages

8.5   View Ribbon Tab

The Windows group, available from the View ribbon tab, contains controls to toggle visibility of the tool windows:

Windows Group

8.5.1   Windows Group
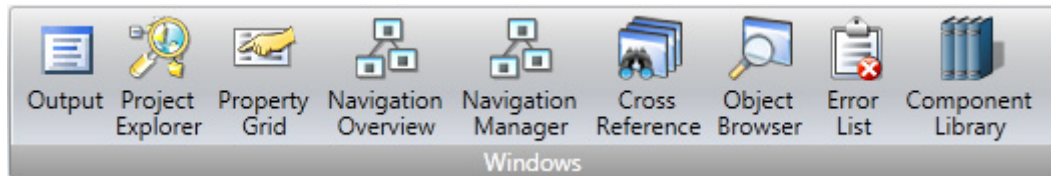
The tool controls in the Windows group are highlighted when active.

Tool window	Description	Default Position
Project Explorer	Shows all screens and components included in the application	Docked to the left part of the application window
Output	Displays detailed information concerning the project validation and build	Docks to the lower part of the application window
Cross Reference	Provides an overview of where a specific tag is used	Docks to the right part of the application window
Error List	Displays information about problems and errors detected during build	Docks to the lower part of the application window
Navigation Overview	Provides an overview of all screens included in the project, with zooming facilities.	Floating window
Navigation Manager	Manages screens and screen navigation	Desktop area
Property Grid	Shows detailed properties of a selected screen or object	Docks to the right part of the application window as a tab
Object Browser	Displays an overview of all objects in the current screen and enables setting visibility and locking objects easily, as well as placing objects in front or back of each other.	Floating window
Component Library	Contains predefined as well as user-defined graphical components	Docks to the right part of the application window as a tab

8.6 Dynamics Ribbon Tab

The Dynamics ribbon tab includes the following groups:

Layout Group
Color Group
General Group

The controls on the Dynamics tab are used to change the properties of an object depending on a changed tag value.

A lock icon resembling a chain indicates that a specific dynamic setting is made for the selected object.

If, for example, dynamic fill settings are made, it is not possible to make fill settings for the object from the Format group on the Home tab.

Note:
Security has higher priority than dynamics.
An object with visibility set to hidden through security cannot have its Visible property changed by dynamics.
An object with visibility set to disabled through security cannot have its IsEnabled property changed by dynamics.

Note:
Scripting to a property for which Dynamics has been configured disables the dynamic settings.

To release an object from a dynamic control, click the control with the lock icon and click **Clear Dynamics** in the appearing dialog.

8.6.1   Layout Group
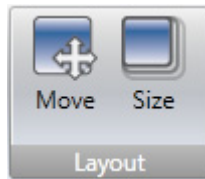
Move

The Move control in the Position group changes an object's position, based on a tag value.

Position coordinates on the screen are given in pixels of the screen resolution. The position 0,0 corresponds to the upper left corner of the screen.

To move an object vertically when a tag value changes:

1. Draw and position the object at the start position. Keep the object selected.
2. Click on **Move** on the Dynamics tab to open the Move Dynamics Editor.
3. Select the tag to control the movement.
4. Enter a start and end value for the tag.

   Start and end positions are suggested based on the current start position. A "ghost object" will appear at the suggested finish position. A guide line that connects the center of the original object with the ghost object appears.
5. Move the ghost object in the screen to its final position.

   The coordinates for the object is updated. An animation of the move appears on the screen. Coordinate values can also be updated directly in the numerical fields (End Top, End Left). The coordinates for start and finish will limit the movement of the object irrespective if the tag value is outside its start and end value.
6. Click **OK** to save and close the Move Dynamics Editor.
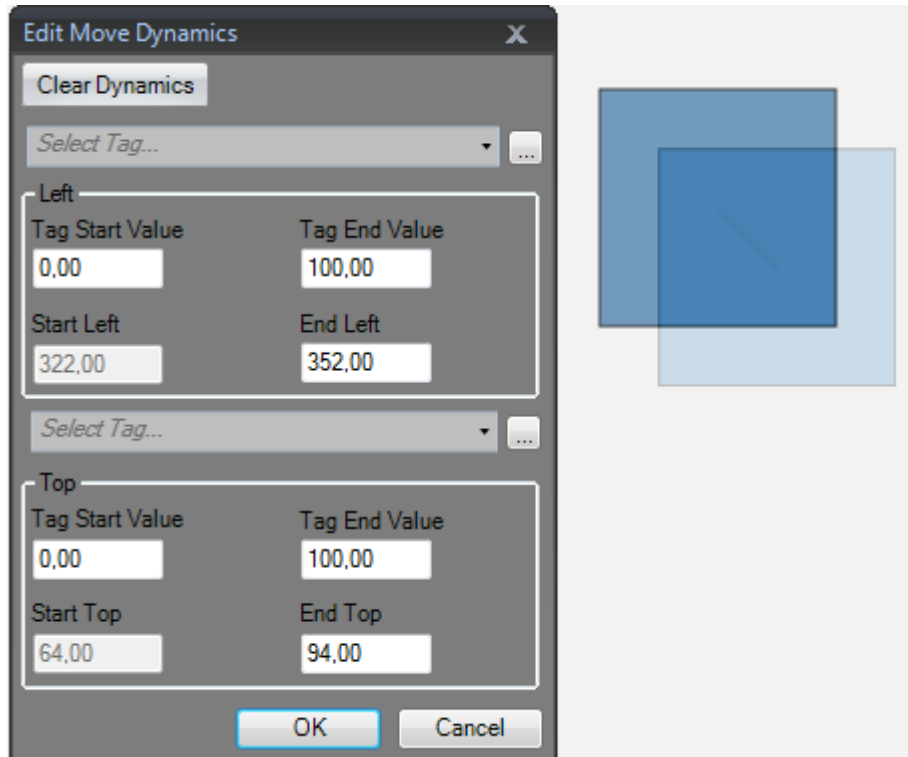
Note:

When Position or Size dynamics is applied, moving or resizing of the object will be disabled. A lock icon resembling a chain in the upper left corner of the object (when selected) indicates that the object has a dynamic control applied and that it cannot be moved or resized.

Size

The Size control in the Size group changes an object's size, based on a tag value.

Dynamic resizing of external picture files may result in a resolution loss if the picture is enlarged to a size larger than what is used elsewhere in the current project.

Width and height of an object are given in pixels of the screen resolution.

To resize the width of an object when a tag value changes:

1. Draw and size the object at its start size. Keep the object selected.

2. Click on **Size** on the Dynamics tab to open the Resize Dynamics Editor.

3. Select the tag to control the resizing.

4. Fill in the values that for start and finish. Size is given in screen resolution pixels.

   Start size is suggested based on the current start size. A finish size is suggested as a fixed increase of the start size horizontally and vertically. A "ghost object" will appear with the finish size.

Note:
The object can only be resized from left to right and top to bottom.

5. Resize the ghost object in the screen to its final size.

   The coordinates for the object is updated. An animation of the resizing appears on the screen. Coordinate values can also be updated directly in the numerical fields (End Width, End Height). The coordinates for start and finish will limit the size of the object irrespective if the tag value is outside its start and end value.

6. Click **OK** to save and close the Resize Dynamics Editor.

Note:
When Position or Size dynamics is applied, moving or resizing of the object will be disabled. A lock icon resembling a chain in the upper left corner of the object (when selected) indicates that the object has a dynamic control applied and that it cannot be moved or resized.

8.6.2   Color Group

The Fill and Outline controls in the Color group change an object's fill color or outline (border) color, based on a tag value.

To alter the fill color of an object when a tag value changes:

1. Draw the object. Keep the object selected.
2. Click on **Fill** on the Dynamics tab to open the Color Dynamics Editor.
3. Select the tag that is to control the color.
4. Fill in values to trigger color change.
5. Select color and gradient from the color menu.
6. Click **OK** to save and close the Color Dynamics Editor.

Color	Start	End
255; 242; 0	0	50
255; 194; 14	51	101
237; 28; 36	102	152
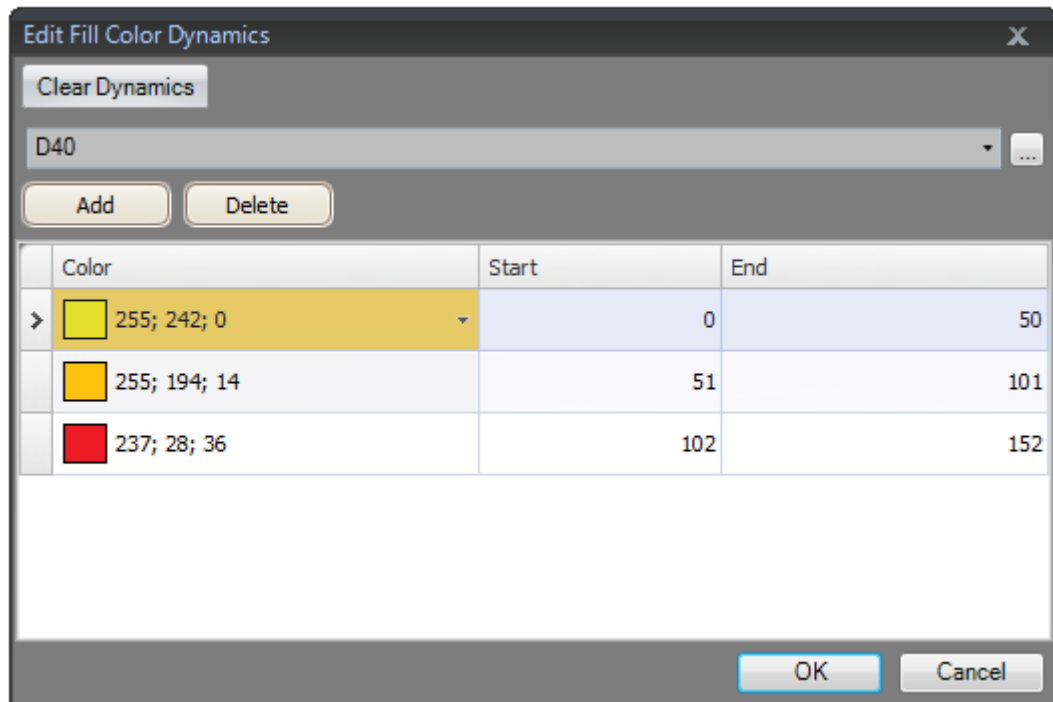
8.6.3   General Group
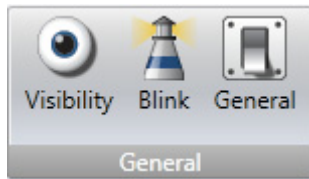
Visability

The Visible control in the Visibility group determines if an object is to be displayed in the screen or not, based on a tag value.

To hide an object when a tag value changes:

1. Draw the object. Keep the object selected.
2. Click on **Visible** on the Dynamics tab.
3. Select the tag that is to control visibility. The object will be visible on the screen at a certain tag value.

Note:
The selected value will be treated as True (visible). All other values will be treated as False (invisible).

Blink

It is possible to select a tag to enable blinking depending on the tag's current value.

Note:
The selected value will be treated as True (blink enabled). All other values will be treated as False (blink disabled).

The blink interval can be set dynamically or constantly.
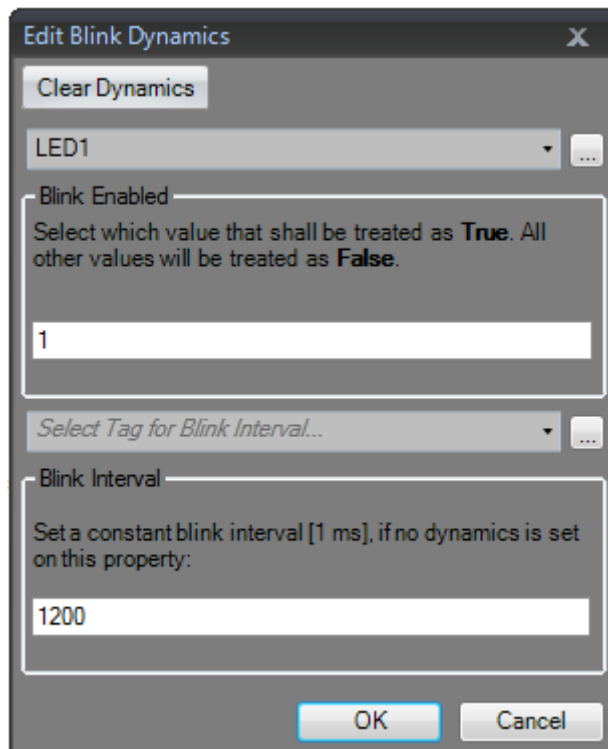
Related information
Blinking Objects

General

The general control offers a possibility to alter the properties of objects in iX Runtime.

Converter Operators

Converter operators are used to alter the properties of the objects. The converter operators can be selected from the **Edit general dynamics** dialog.

Parameter	Description
Bool Converter	Converts 0 to false and all other values to true.
Color Converter	Converts value intervals to specified colors.
Linear Converter	Converts values according to the linear equation.
Raw Converter	Converts the value to a new value matching the target type.
Text Library Group converter	Converts a value to a text.

Edit General Dynamics

The **Edit general dynamics** dialog contains different properties depending on which object is selected. The dialog is accessed by clicking the **General** control on the **General** group.
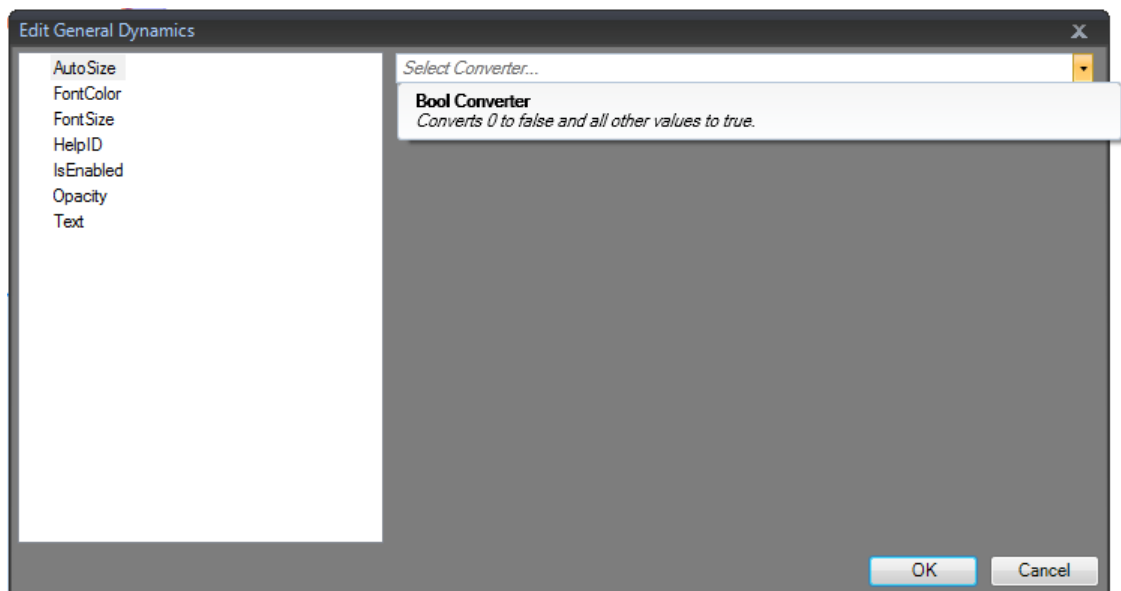
Parameter	Description
AcknowledgeAllButtonIsVisible	The bool converter is used to make the AcknowledgeAll button on the alarm viewer object either visible or not.
AcknowledgeSelectedButtonIsVisible	The bool converter is used to make the AcknowledgeSelected button on the alarm viewer object either visible or not.
Address	The raw converter or a text library group converter can be used to change the default URL in the web browser object.
AlternateBackground	The color interval converter can be used to alter the background color.
AlternateForeground	The color interval converter can be used to alter the foreground color.
AnimationSpeed	The linear converter or the raw converter can be used to alter the animation speed of the animated label object.
AutoSize	The bool converter can be used to resize the object according to the size of the default text string length and font size.
ButtonColor	The color interval converter can be used to alter the color of a buttons in the alarm viewer object.
ButtonHeight	The linear converter or the raw converter can be used to alter the height of the buttons in the alarm viewer object.
ButtonWidth	The linear converter or the raw converter can be used to alter the width of the buttons in the alarm viewer object.
ClearButtonIsVisible	The bool converter is used to make the Clear button on the alarm viewer object either visible or not.
Clockwise	The bool converter is used to alter the rotation orientation in the circular meter object.
ColumnHeaderBackgroundColor	The color interval converter can be used to alter the column header background color.
EndAngle	The linear converter or the raw converter can be used to alter the end angle in the circular meter object.
FileName	The raw converter or a text library group converter can be used to select a PDF file to add to the PDF viewer.
FilterButtonIsVisible	The bool converter is used to make the filter button on the alarm viewer object either visible or not.

Parameter	Description
FontColor	The color interval converter can be used to alter the font color in for instance the text object.
FontSize	The linear converter or the raw converter can be used to alter the font size in for instance the text object.
GroupBackground	The color interval converter can be used to alter the group background color in the action menu object.
GroupForeground	The color interval converter can be used to alter the group foreground color in the action menu object.
Header	The raw converter or a text library group converter can be used to change the header in the chart object.
HighRegionColor	The color interval converter can be used to alter the high region color in the circular meter object.
HighRegionMaximum	The linear converter or the raw converter can be used to alter the maximum value of the high region indicator in the circular meter object.
HighRegionMinimum	The linear converter or the raw converter can be used to alter the minimum value of the high region indicator in the circular meter object.
IndicatorColor	The color interval converter can be used to alter the indicator color in for instance the circular meter object.
InfoButtonIsVisible	The bool converter is used to make the info button on the alarm viewer object either visible or not.
IsActive	The bool converter can be used to activate or stop the data input to the chart object.
IsAnimated	The bool converter can be used to enable or disable the animation of the animation label object.
IsEnabled	The bool converter can be used to enable or disable an object.
IsHeaderVisible	The bool converter can be used to make the header on the chart object visible or not.
IsLegendVisible	The bool converter can be used to make the legend on the chart object visible or not.

Parameter	Description
IsReadOnly	The bool converter can be used to make for instance an analog numeric object editable or not.
IsXAxisAutomatic	The bool converter can be used with the chart object to toggle whether iX Runtime automatically should calculate the minimum and maximum X Axis values or not.
IsXAxisGridVisible	The bool converter can be used with the chart object to toggle the visibility of the grid.
IsXAxisVisible	The bool converter can be used with the chart object to toggle the visibility of the X Axis.
IsY1AxisAutomatic	The bool converter can be used with the chart object to toggle whether iX Runtime automatically should calculate the minimum and maximum Y1 Axis values or not.
IsY1AxisGridVisible	The bool converter can be used with the chart object to toggle the visibility of the grid.
IsY1AxisVisible	The bool converter can be used with the chart object to toggle the visibility of the Y1 Axis.
IsY2AxisAutomatic	The bool converter can be used with the chart object to toggle whether iX Runtime automatically should calculate the minimum and maximum Y2 Axis values or not.
IsY2AxisGridVisible	The bool converter can be used with the chart object to toggle the visibility of the grid.
IsY2AxisVisible	The bool converter can be used with the chart object to toggle the visibility of the Y2 Axis.
LockAspectRatio	The bool converter can be used with the multi picture object to toggle the height / width ratio of the object.
LowRegionColor	The color interval converter can be used to alter the low region color in the circular meter object.
LowRegionMaximum	The linear converter or the raw converter can be used to alter the maximum value of the low region indicator in the circular meter object.

Parameter	Description
LowRegionMinimum	The linear converter or the raw converter can be used to alter the minimum value of the low region indicator in the circular meter object.
MajorTickCount	The linear converter or the raw converter can be used to set the number of labeled time marks in for instance the linear meter object.
Maximum	The linear converter or the raw converter can be used to alter the end value of the scale in for instance the linear meter object.
MaxValue	The linear converter or the raw converter can be used to alter the maximum value that can be entered in the analog numeric object.
MediumRegionColor	The color interval converter can be used to alter the medium region color in the circular meter object.
MediumRegionMaximum	The linear converter or the raw converter can be used to alter the maximum value of the medium region indicator in the circular meter object.
MediumRegionMinimum	The linear converter or the raw converter can be used to alter the minimum value of the medium region indicator in the circular meter object.
Minimum	The linear converter or the raw converter can be used to alter the start value of the scale in for instance the linear meter object.
MinorTickCount	The linear converter or the raw converter can be used to set the number of scale ticks between two adjacent major ticks.
MinValue	The linear converter or the raw converter can be used to alter the minimum value that can be entered in the analog numeric object.
Opacity	The linear converter or the raw converter can be used to alter the opacity of for instance the PDF viewer object.
Outline Thickness	The linear converter or the raw converter can be used to alter the outline thickness of for instance the rectangle object.
PauseButtonIsVisible	The bool converter is used to make the pause button on the alarm viewer object either visible or not.

Parameter	Description
PictureHeight	The linear converter or the raw converter can be used to alter the picture height in the button object.
PictureName	The raw converter or a text library group converter can be used to change the picture shown in the picture object.
PictureStretch	The bool converter is used to stretch a picture used in the button object.
PictureWidth	The linear converter or the raw converter can be used to alter the picture width in the button object.
Prefix	The raw converter or a text library group converter can be used to set a prefix to the digits entered in the analog numeric object.
RadiusX	The linear converter or the raw converter can be used to alter the radius of the x axis in the rectangle object.
RadiusY	The linear converter or the raw converter can be used to alter the radius of the y axis in the rectangle object.
ScaleColor	The color interval converter can be used to alter the scale color in the circular meter object.
ScaleVisible	The bool converter is used to show or hide the scale in the circular meter object.
SelectedBackground	The color interval converter can be used to alter the background color of the selected action in the action menu object.
SelectedForground	The color interval converter can be used to alter the foreground color of the selected action in the action menu object.
SelectedGroupBackground	The color interval converter can be used to alter the background color of the selected menu group in the action menu object.
SelectedGroupForeground	The color interval converter can be used to alter the foreground color of the selected menu group in the action menu object.
SeparatorColor	The color interval converter can be used to alter the separator color displayed between the buttons in the touch list object.

Parameter	Description
Source	The raw converter or a text library group converter can be used to set the source file for the media player object.
StartAngle	The linear converter or the raw converter can be used to alter the start angle in the circular meter object.
Stretch	The bool converter can be used to adjust the height and width to the selection frame in for instance the picture object.
Suffix	The raw converter or a text library group converter can be used to set a suffix to the digits entered in the analog numeric object.
Text	The raw converter or a text library group converter can be used to alter the text shown in for instance the text object.
Title	The raw converter or a text library group converter can be used to alter the title of the roller panel object.
TitleBackground	The color interval converter can be used to alter the title background color in the roller panel object.
TitleForeground	The color interval converter can be used to alter the title foreground color in the roller panel object.
Transparent	The bool converter can be used to toggle the transparency of for instance the button object.
Value	The linear converter or the raw converter can be used to alter the value in the progress bar object.
ValueScaleMajorTickCount	The linear converter or the raw converter can be used to alter the number of labeled scale marks on the scale of the trend viewer object.
ValueScaleMaximum	The linear converter or the raw converter can be used to alter the end value of the scale in the trend viewer object.
ValueScaleMinimum	The linear converter or the raw converter can be used to alter the start value of the scale in the trend viewer object.
ValueScaleMinorTickCount	The linear converter or the raw converter can be used to alter the number of scale ticks between two adjacent major ticks on the scale of trend viewer object.

Parameter	Description
X1	The linear converter or the raw converter can be used to alter the X1 pixel coordinates of the line object.
X2	The linear converter or the raw converter can be used to alter the X2 pixel coordinates of the line object.
X3	The linear converter or the raw converter can be used to alter the X3 pixel coordinates of the line object.
X4	The linear converter or the raw converter can be used to alter the X4 pixel coordinates of the line object.
XAxisGridColor	The color interval converter can be used to alter the grid color of the X axis in the chart object.
XAxisMaximum	The linear converter or the raw converter can be used to alter the maximum value of the X axis in the chart object.
Y1AxisGridColor	The color interval converter can be used to alter the grid color of the Y1 axis in the chart object.
Y1AxisMaximum	The linear converter or the raw converter can be used to alter the maximum value of the Y1 axis in the chart object.
Y1AxisMinimum	The linear converter or the raw converter can be used to alter the minimum value of the Y1 axis in the chart object.
Y2AxisGridColor	The color interval converter can be used to alter the grid color of the Y2 axis in the chart object.
Y2AxisMaximum	The linear converter or the raw converter can be used to alter the maximum value of the Y2 axis in the chart object.
Y2AxisMinimum	The linear converter or the raw converter can be used to alter the minimum value of the Y2 axis in the chart object.

Related information
Text Library

8.7 General Ribbon Tab

The General ribbon tab contains different property groups and controls for different objects. Appearance can be modified for graphical shapes and objects. Scale settings can be made for meters and trend viewers. Additional settings can be made for some objects.

The Tag/Security group is available also on the Home ribbon tab.

The general properties are described separately for each object.

Related information
HMI Controls in the *Objects* group
Media Controls in the *Objects* group
Special Controls in the *Objects* group
Windows Controls in the *Objects* group
Tag / Security Group on the *Home Ribbon Tab*

8.8 Actions Ribbon Tab

Actions can be triggered in various ways, for example by clicking a button. The Actions tab controls are used to define actions for objects. Alternatively, a script can be used to trigger an action.

An action is something that is performed, for example closing a screen. An action trigger is the act that generates an action, for example clicking a button.

The Actions ribbon tab contains different control groups for different objects.
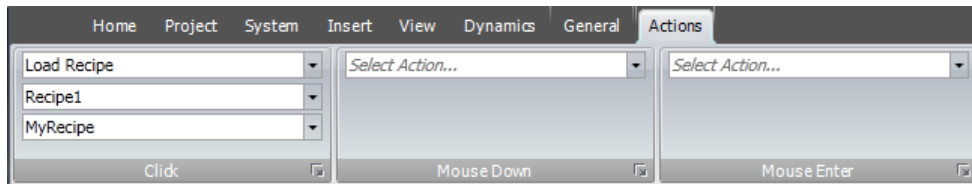
A Click action configured for a button

Note:
It is not recommended to use multiple trigger methods, as this may lead to undesired behavior. It is recommended either to use a script or to program actions with the controls in the Actions tab. For example, avoid using *Click Action Trigger* in combination with *Mouse Button Action Triggers*.

Related information
Script

8.8.1   Click Action Trigger

Most objects can be assigned click actions, that will be executed when the object is clicked in runtime.

8.8.2   Mouse Button Action Triggers

The Mouse Down and Mouse Up action triggers can be used to create two-step actions when clicking with the left mouse button on an object.

Parameter	Description
Mouse Down	A Mouse Down action will be performed when the left mouse button is pressed down on the object in runtime.
Mouse Up	A Mouse Up action will be performed when the left mouse button is released from the object in runtime.
Mouse Enter	A Mouse Enter action will be performed when the cursor enters the boundaries of an object in runtime.
Mouse Leave	A Mouse Leave action will be performed when the cursor leaves the boundaries of an object in runtime.

8.8.3   Function Key Action Triggers

The Key Down and Key Up action triggers can be used to create two-step actions for function keys.

Parameter	Description
Key Down	A Key Down action will be performed when a function key is pressed in runtime.
Key Up	A Key Up action will be performed when a function key is released in runtime.

8.8.4   Value Changed Action Triggers

Parameter	Description
Input Value Changed	The Input Value Changed action will be performed when a value is entered by the operator.
Value Changed	The Value Changed action will be performed when the value of the object has changed.

8.8.5   Focus Action Triggers

Parameter	Description
Got Focus	The Got Focus action will be performed when the object is selected.
Lost Focus	The Lost Focus action will be performed when the object is no longer selected.

8.8.6 Navigation Action Triggers

Parameter	Description
Navigated	The Navigated action will be performed when having navigated to the address entered in the address field of the Web browser object.
Navigating	The Navigating action will be performed when having entered, but not reached, an address in the address field of the Web browser object.

8.8.7 Data Logger Action Triggers

The LogItemValueChanged and LogValuesUpdated are available for the Data Logger.

8.8.8 Alarm Server Action Triggers

Alarm server actions can be configured for the entire alarm server, for alarm groups or for individual alarm items. The actions will be activated upon alarm status changes.

Related information
Alarm Server Events and Actions
Actions and Events for Alarm Items and Alarm Groups

8.8.9   Action Groups

Actions are divided into action groups.

| Action group	Action	Description
Screen	Close Screen	Closes the current screen
	Print Screen	Prints the current screen on printer or to a .PNG file. Selecting Print to file enables the following options: **Project files folder**: The file will be saved in a folder named "Project Files" in the projects execution path. **USB**: The file will be saved to a connected USB stick. (Operator panels only). **Memory card**: The file will be saved to the external memory card. (Operator panels only). **Browse**: A save dialog will be shown in runtime. (PC only). **Path**: The file will be saved to a specified path. (PC only).
	Show Next Screen	Shows next screen
	Show Previous Screen	Returns to previous screen
	Show Screen	Shows the specified screen. If the screen is a popup the position of the screen can be defined by entering (X,Y) coordinates.
	Show Start Screen	Shows the screen set to Startup screen
Address Book	Open Address Book	Opens the address book for editing. This action is used for *Alarm Distributor* purposes.
Alarm Distributor Server	Open Print Settings	Modifies alarm distributor print settings
	Open Routes Configuration	Modifies routes for the distributed alarm server
Audit Trail	Export Audit Trail Database	Exports the audit trail database to an USB stick (Operator Panel) or to a folder (PC). When this is done the audit trail log is cleared. The export will be saved in a .SDF file.

Action group	Action	Description
Chart	Pan Down	Pans down in selected chart
	Pan Left	Pans left in selected chart
	Pan Right	Pans right in selected chart
	Pan Up	Pans up in selected chart
	Reset View	Resets any panning and zoom in selected chart
	Zoom In	Zooms into selected chart
	Zoom Out	Zooms out of selected chart
Controller	Change Active Controllers	Defines which controllers are to be active after having restarted the project
	Clear Non-Volatile Values	Clears all stored values for tags marked as non-volatile
	Decrement Analog	Decrements tag value This action can be used with *Aliases*.
	Increment Analog	Increments tag value This action can be used with *Aliases*.
	Reset Tag	Resets tag to 0 This action can be used with *Aliases*.
	Set Analog	Sets tag to the specified value This action can be used with *Aliases*.
	Set String	Sets tag to the specified string This action can be used with *Aliases*.
	Set Tag	Sets tag to 1 This action can be used with *Aliases*.
	Toggle Tag	Toggles tag between 0 and 1 This action can be used with *Aliases*.
Database	Backup Database	Performs a backup of the project database
	Database Export	Exports selected database in csv format
Datalogger	Clear Datalogger	Clears the specified datalogger
	Log Once	Makes the specified datalogger log once
	Start Logging	Enables the specified datalogger
	Stop Logging	Disables the specified datalogger
Output Devices	Email Configuration	Shows the e-mail configuration page. This action is used for *Alarm Distributor* purposes.
	Printer Configuration	Shows the printer configuration page.
	SMS Configuration	Shows the SMS configuration page. This action is used for *Alarm Distributor* purposes.

Action group	Action	Description
Recipe	Delete Recipe	Deletes recipe
	End Offline Recipe Editing	Sets the tags in the specified recipe in online mode.
	Export Recipe	Exports a recipe to a .CSV file
	Import Recipe	Imports a recipe from a .CSV file
	Load Recipe	Loads recipe
	Save Recipe	Saves recipe
	Start Offline Recipe Editing	Sets the tags in the specified recipe in offline mode.
Reporting	Generate Report	Generates a report file that can be output to a printer or saved as an Excel file. Selecting "File" enables the following options: **Project files folder**: The file will be saved in a subfolder named "Reports" in the "Project Files" folder. **USB**: The file will be saved to a connected USB stick. (Operator panels only). **Memory card**: The file will be saved to the external memory card. (Operator panels only). **Browse**: A save dialog will be shown in runtime. (PC only). **Path**: The file will be saved to a specified path. (PC only).
Security	Export User Accounts	Exports user data to a location that the operator states
	Import User Accounts	Imports user data from a location that the user states
	Login	Displays a login dialog
	Logout	Logs out the current user
	Show Users Dialog	Displays a dialog where users can be added, edited or removed
Trend Viewer	Show Trend Viewer Legend	Displays the legend tool with a possibility to show and hide curves, and to display values of a specific trend viewer sample.
	Trend Viewer History	The Trend Viewer History action includes the following choices: **On**: Shows trend history from when the action was triggered (including offset settings made using trend viewer dynamics). **Off**: Returns to showing real time trend viewer data.

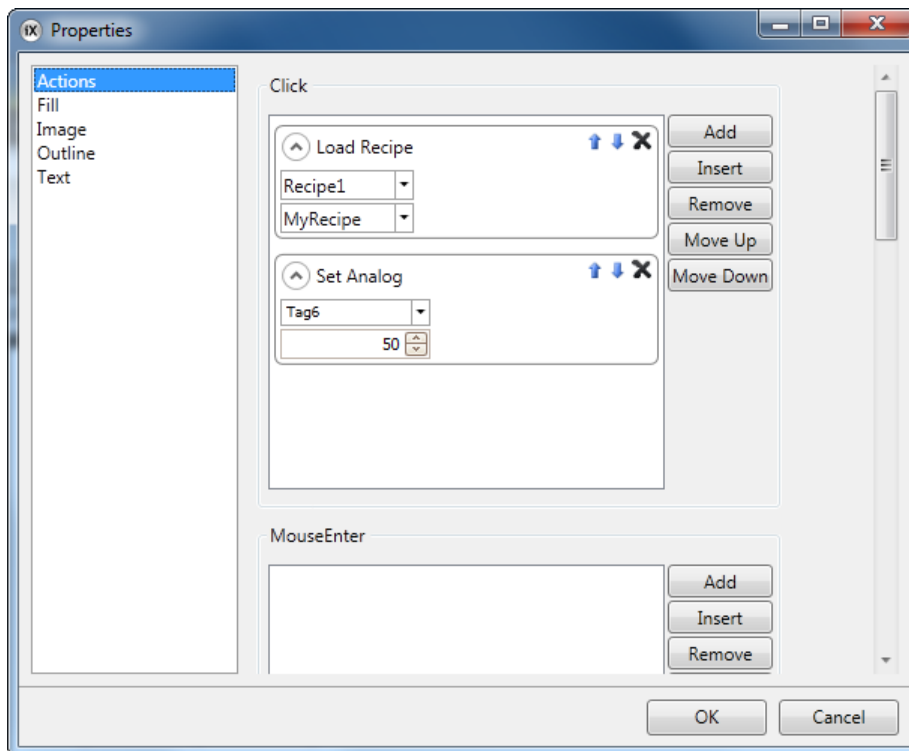| Action group	Action	Description
		Toggle: Allows entering and terminating history mode using the same object (for example a button). **Show Dialog**: Lets the user select a period of time, for which to show trend viewer history.
Other	Close Application	Closes the runtime application in a PC project. In an operator panel, the action closes the application, and then the application is restarted by the shell afterwards.
	Copy Debug Log to USB Memory	Copies the debug log to a USB memory.
	Disable Debug Logger	Disables the debug logger.
	Enable Debug Logger	Enables the debug logger.
	Run	Starts an external application
	Scroll Left	Scrolls the screen carousel object left
	Scroll Right	Scrolls the screen carousel object right
	Set Date and Time	Allows changing date and time in the operator panel
	Set Language	Sets the language in the operator panel to the selected language
	Set Time Zone, Region and Daylight Saving	Allows changing time zone, region and daylight saving settings in the operator panel
	Show Backlight Settings	Allows changing backlight settings in the Operator panel
	Show IP Settings	Shows the IP settings in the Operator panel

8.8.10   Multiple Actions

Multiple actions can be configured by clicking small arrow in the lower right corner of any action group.
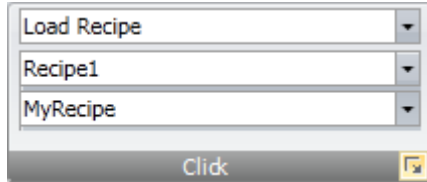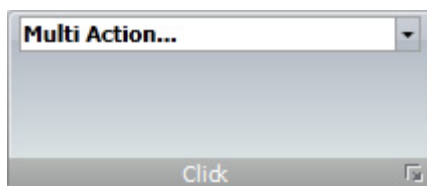

Note the small arrow in the lower right corner

The properties window allows configuring and sorting additional actions. The actions available depend of which object is selected in the screen prior to selecting the Actions ribbon tab.

After configuring multiple actions, this is indicated for the action group on the Actions tab.

9 Trend Viewer

This section describes the trend viewer object. Trend Viewers are used to present chronological data from controller tags. The curves in a trend viewer can have different sources of data.

The source of data can be a log item for a data logger. The values of the tag connected to the log item is then stored in the projects database.

The source can also be a Tag. The trend viewer curve then only shows real-time values and is stored in a RAM cache.

Related information
Database
Database Export

9.1 Defining Trend Viewer Objects

It is possible to define several independent trend viewer objects, in the same screen or in different screens. The number of trend viewer curves is limited only by the available amount of memory in the panel.

Note:
A large number of trend viewer pens and short sampling intervals may affect communication performance.

Related information
Communication Performance

9.1.1 Adding a Trend Viewer Object

A trend viewer object is added to a screen from the Objects group on the Home ribbon tab.

Settings for the trend viewer object are available from the General ribbon tab while the trend viewer object is selected:
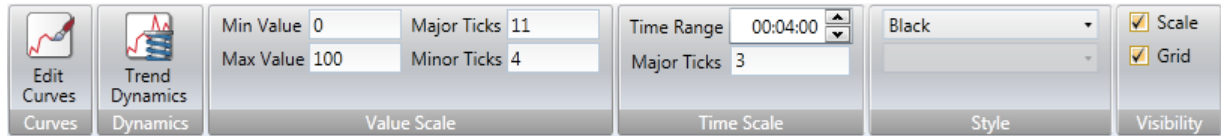
Curves Group

Trend viewer curves are defined using the **Edit Curves** control.

Parameter	Description
Add	Adds a new curve to the trend viewer object
Remove	Removes the selected curve from the trend viewer object
Name	The symbolic name of the curve. This name is shown if a *Trend Viewer Legend* is used.
Tag	The tag to present values for
Log Item	The item to be logged
Expression	Expression on a curve is evaluated and applied when the curve is drawn. If the expression value when the tag is logged or cached is of interest, the expression on the log item or the tag can be used. See section *Expressions*
Color	The color of the curve
Thickness	Line thickness
Minimum Value Tag	A tag that dynamically sets what value on the curve that corresponds to the minimum value of the Y axis. This property will scale the curve when the value of the tag is different from the minimum value of the Y axis (optional).
Maximum Value Tag	The tag that dynamically sets what value on the curve that corresponds to the maximum value of the Y axis. This property will scale the curve when the value of the tag is different from the maximum value of the Y axis (optional)

Dynamics Group

Several trend viewer properties can be made dynamic, i. e. dependent on tag values. This may be useful, for example, when wanting to present trend viewer values of different formats in the same trend viewer object.

If dynamic trend viewer properties are used, the corresponding property settings made in the Value Scale group for the trend viewer object are ignored.

Parameter	Description
Value Scale Min / Value Scale Max	Minimum/maximum value of the Y-scale. The design mode value is used only for increased understanding in design time when tag values do not affect the minimum and maximum values.
Time Span	The time span of the X-axis in seconds.
Time Offset in History Mode	Makes it possible to use an offset to step back in history mode. The start value is when history mode was entered. The unit is seconds.

Value Scale Group

The value scale group controls the vertical scale (y-axis). If dynamic trend viewer properties are used, the settings made for these properties in the value scale group will be ignored.

Parameter	Description
Min Value/Max Value	Start/end value of scale
Major Ticks	Number of labeled scale marks on the scale
Minor Ticks	Number of scale ticks between two adjacent major ticks

Time Scale Group

Parameter	Description
Time Range	Visible time span for a trend viewer in Hour:Minute:Second format
Major Ticks	Number of labeled time marks. This property will be ignored in *Historical Mode in Trend Viewer*.

Style Group

Parameter	Description
Style	Predefined graphical styles for the trend viewer

Visibility Group

Parameter	Description
Scale	When unchecked, the scale color of the trend is made transparent. Scale color is set from **Other Colors** in the Format group on the Home ribbon tab.
Grid	When unchecked, the grid color of the trend viewer is made transparent. Grid color is set from **Other Colors** in the Format group on the Home ribbon tab.

9.1.2   Trend Viewer Legend

In runtime, it is possible to display a trend viewer legend tool using the Show Trend Viewer Legend action. The trend viewer legend tool is a floating window that displays the name and color of the curves in the current trend viewer object. Curves can be made invisible by clicking the visibility icon.

Use the needle directly in the trend viewer to select a specific occasion. The values of all curves at the current position are displayed in the trend viewer legend tool.
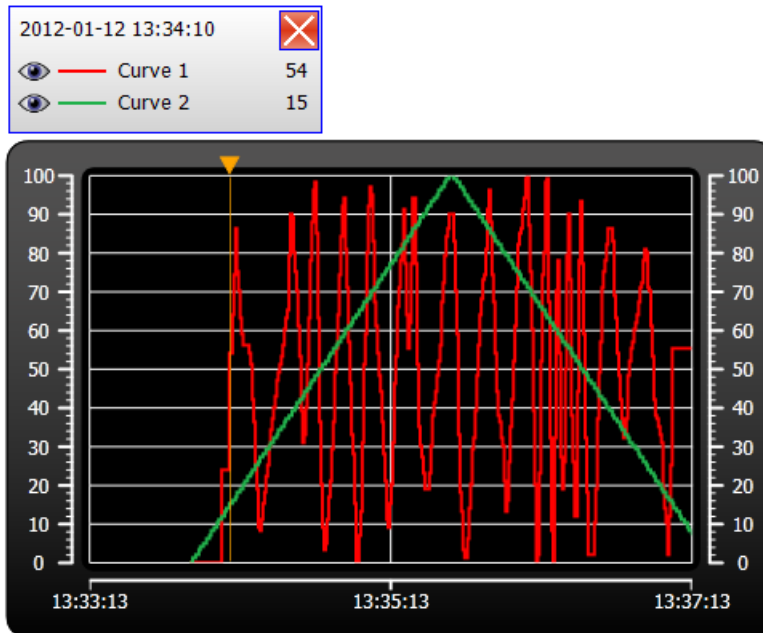
9.2 Historical Mode in Trend Viewer

To be able to show historical data in a trend viewer, the curves needs to be connected to log items from a data logger.

To show trend viewer history, it is necessary to define trend actions to show and close the historical trend viewer. These actions can for example be programmed as button click actions. It is also possible to let the user select a time period.

Major ticks settings for time axis will be ignored in historical trend viewers. In historical trend viewers the date and time will be viewed at start and end time. The text "history" is visible in the middle of the time axis.
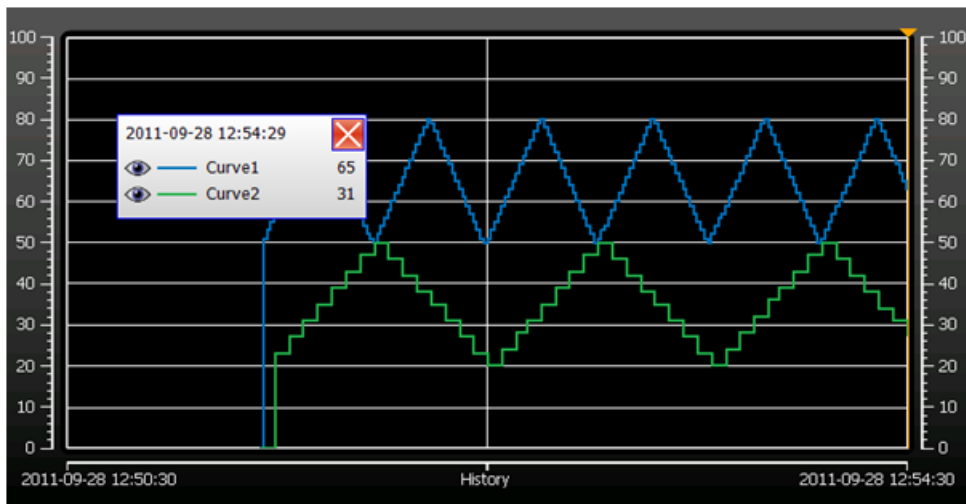
Related information
Performance in the Operator Panel
Database
Click Action Trigger

10   Data Logger

Data can be logged and saved in the iX Developer database.

Any number of tags can be connected to a data logger, and data values irrespective of type can be logged. A project can have multiple data loggers with different sampling alternatives.

When installing iX Developer, a Database Viewer is automatically installed and available from Windows Start menu. The Database Viewer makes it possible to open the database file and inspect logged data, and to export the file to Excel.

Logged data can also be viewed in the trend viewer object by selecting a LogItem for a curve in the Edit Curves dialog.

Note:
Logging of data consumes system resources and memory. To save memory, it is preferable that all LogItems that should be logged with the same sampling intervals are placed in the same data logger.

Any individual data logger database can be exported in csv format using the **Database Export** action. It is recommended to use the database export method rather than the backup method.

Related information
Database
Database Export
Adding a Trend Viewer Object

10.1   Data Logging Strategies

Logging of data can be made at time intervals or based on changed tag values.

Logging of data can be made either by collecting all samples at every log event, or by collecting only changed values. Collecting only changed values saves memory and improves performance.

### 10.1.1   Logging Based on Time Interval

To collect data that changes frequently, logging data based on time interval is recommended.

10.1.2   Logging Based on Changed Tag Value

To control sampling of data by other means than regular time interval, you have to set up a separate tag to be used as a trigger. When the value of the trigger tag changes the log item is logged.
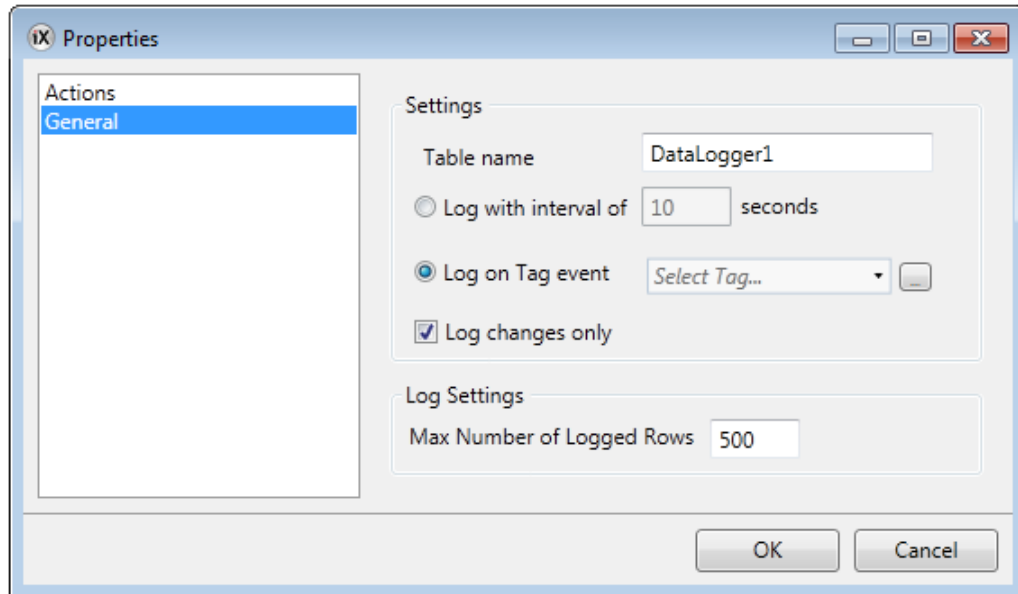
10.2 Adding a Data Logger

A data logger can be added to the project from the Insert ribbon tab. The data logger configuration pages are used to set up which values to log.
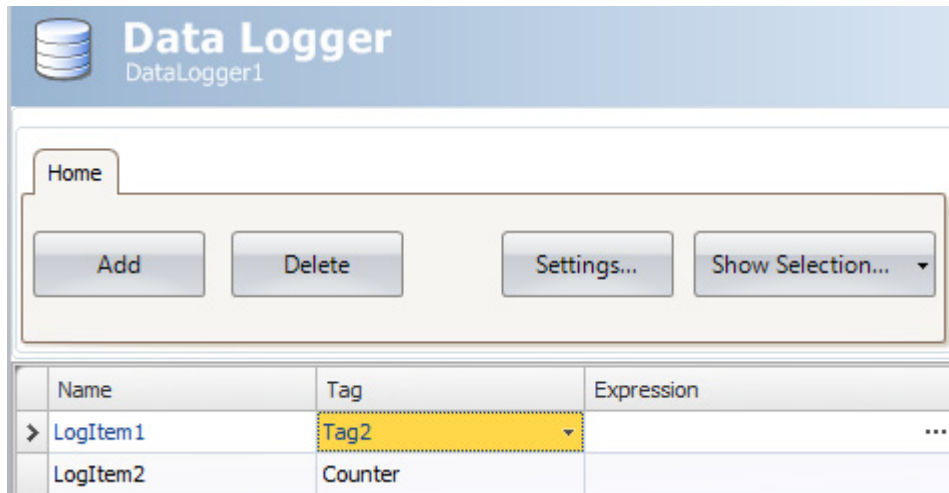
Parameter	Description
Name	Symbolic name for the logged tag
Tag	Tag to log
Expression	See section *Expressions*

Clicking the **Settings** button accesses the Data Logger settings.

Related information
General Settings
Actions

10.2.1   General Settings

Parameter	Description
Table name	The name of the data logger table. This name is the same as the data logger name.
Log with interval	To collect data that changes frequently, use the **Log with interval of** option, and define time interval in seconds. Minimum interval is 0.1 second for PC projects and 1 second for panel projects.
Log on object event	To control sampling of data by other means than regular time interval, select the **Log on object event** option, and select the tag to use as trigger. Data will be sampled when the tag value changes.
Log changes only	By logging changes only saves memory. The option **Log changes only** can be combined with either the **"Log with interval of"** or the **"Log on object event"** option. Data will only be logged if the value differs from the last logged value.
Max number of logged rows	The maximum number of tag values to save in the database. When the number of collected items becomes larger than the set value, the oldest item will be overwritten. The value must be within the size of the available amount of memory in the operator panel in runtime.

Related information
Database
Click Action Trigger

10.2.2   Actions

Event	Description
LogItemValueChanged	When a value of an particular log item has changed
LogValuesUpdated	When any log value is updated

11   Scheduler

A scheduler is used to control tags in relation to the real-time clock. This function is used to control events in the process, for example starting and stopping motors, at specific predefined calendar times.
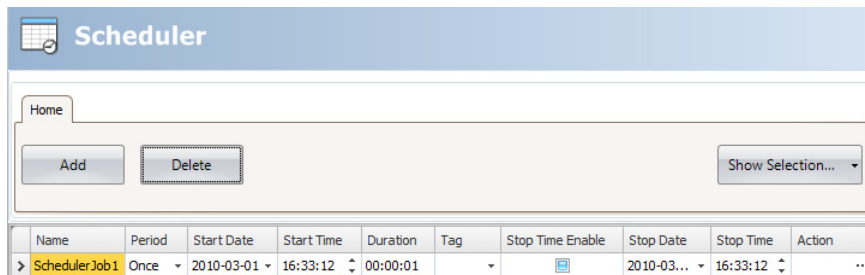
11.1 Scheduler Set Up

The scheduler configuration pages are used to set up scheduler jobs, which can include a set of tags with associated start and stop times.

11.1.1   Adding a Scheduler

A scheduler can be added to the project from the Insert ribbon tab.

The scheduler configuration page will open in the desktop when a scheduler function is created or selected.

Parameter	Description
Name	A symbolic name for the scheduled event.
Period	Set to Once (default) or to a regular interval (every minute, hour, day, week, month or year).
Start Date/Start Time	The date and time for the first scheduled event.
Duration	The length of the event.
Tag	A digital tag, which is set to 1 during the specified interval.
Stop Time Enable	When checked, Stop Date and Stop Time, instead of Duration, can be used to determine the end of the event.
Stop Date/Stop Time	The end date and time of the event. Also resets the defined Tag to 0. Used when Stop Time Enable is checked.
Action	Click the button to configure an action to be performed when the scheduled event is activated/deactivated.

Note:
Make sure not to use the same trigger tag for two different schedulers. When the first scheduler is finished, its trigger tag is reset, and the second scheduler will not be triggered.

12   Reports

The report generator allows generating Excel reports based on an Microsoft Excel made report template.

The generated report can be output to a printer or saved as an Excel file.

12.1 Reports Template Set up

The tag data to be displayed in the report file are set up in the template file using "placeholders".

The placeholders acts as substitute for the actual tag data that is to be inserted when the report is generated in iX Runtime.

To add a placeholder, enter the following text inside a cell in the Excel report template:

<#Tag(TagName)>

Replace "TagName" with the actual name of the tag to be displayed.

When the template file is complete it needs to be added to the iX Developer project.

Note:
The tag placeholders can only be used in the cells of the Excel template file, they can not be used in, for example, Microsoft Excel WordArt texts.

12.1.1   Limitations

- Microsoft Excel xlsx format is not supported by the report generator.
- Reports containing charts and pictures can not be printed from a panel target.
- The report generation is not an instant process. It may elapse a shorter period of time between the first sampled tag and the last sampled tag, depending on the number of tags to process.

12.1.2   Database Driven Reports

To be able to create database driven reports, the report template must be prepared as described in these instructions:

Configuration Sheet

The configuration sheet is a repository where configurations on the Microsoft Excel report template can be made. This sheet will not be visible in the generated report.

1. Create a new worksheet in the report template. Name the sheet <#Config>.

2. Row 10, and all rows below, in column A and B can be used to configure the SQL queries. Enter the SQL query in column B. Use the corresponding cell in column A to enter a name for the query. The name will be used to refer to the query in the report template.

Enter the SQL queries in the following format:

SQL(DatabaseName; SQL query)

Example: SQL(General; SELECT * FROM DataLogger1)

There are two available databases: General and AuditTrail.

Present the Database Data in the Report

The data retrieved from the SQL database, using the queries in the configuration sheet, can be presented in the report using "named ranges" in Microsoft Excel.

A named range is created in Microsoft Excel by following these steps:

1. Select the cells to be included in a named range.

2. Click the **Define name** command from the **Insert** menu. (In Excel 2007, go to the **Formulas** tab and choose **Name Manager**).

3. Enter a name for the range. The name must be entered in the following format:
   "__QueryName__".
   Replace "QueryName" with the name that you have chosen for your SQL query.

To present the data in the report, enter the following syntax into the cells included in the named range:

<#QueryName.DatabaseColumnName>

Replace the "DatabaseColumnName" entry with the actual name of the table in the SQL database that you wish to present in the report.

Note:

It is possible to use the named range cells in Microsoft Excel charts and functions.

To be able to use the named range in functions, an empty row must be inserted under the named range cells. This empty row must also be included in the function.

Related information
Adding a Report

12.2 Reports Set Up

The reports configuration page are available from the Insert ribbon tab.
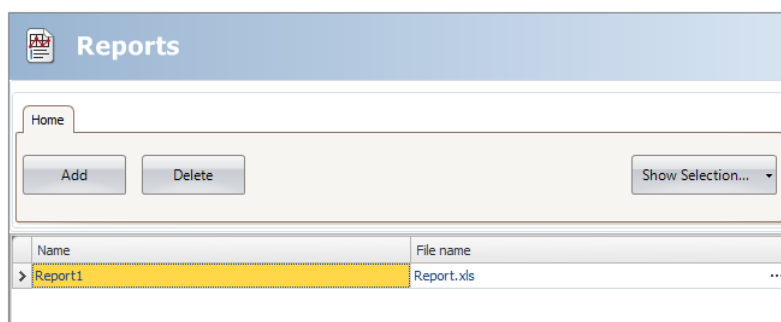
Related information
Insert Ribbon Tab

12.3 Adding a Report

Add a Microsoft Excel report template by pressing **Add** in the reports configuration page. In the appearing Add report dialog, type in or browse to the location of the Excel template file.

If any changes are made to a report template after it has been uploaded to the project, the template has to be uploaded again.

The template files being added to the project must have unique file names.

Parameter	Description
Name	A symbolic name for the report.
File name	The name of the Microsoft Excel report template.

The **Show Selection/Show All** button can be used to set up a filter of displayed items.

Related information
Performance in the Operator Panel
Filtering Items
Configuration Pages
Action Groups
Output Devices Group

13 Recipe Management

Recipes make it possible for a set of tags to be saved in the operator panel to be downloaded at later time. The operator can download a saved recipe to the controller, which will start working with the new values. Recipe libraries consisting of recipes with different sets of parameters can be created, allowing reuse of large sets of parameters.

Recipes can be created during the design of the project or from the operator panel.

Recipe data is stored in the operator panel database. Recipe handling functions such as loading, storing, copying and deleting recipes are handled with actions.

| Related information |
|---|
| *Database* |
| *Actions Ribbon Tab* |

13.1 Recipe Setup

Recipes consist of a set of recipe tags and of recipe data (values).

13.1.1 Adding a Recipe Manager

The recipe function is added to the project from the Insert ribbon tab. The recipe configuration pages are used to manage tag sets and values via the Tag Configuration tab and the Runtime Data tab.
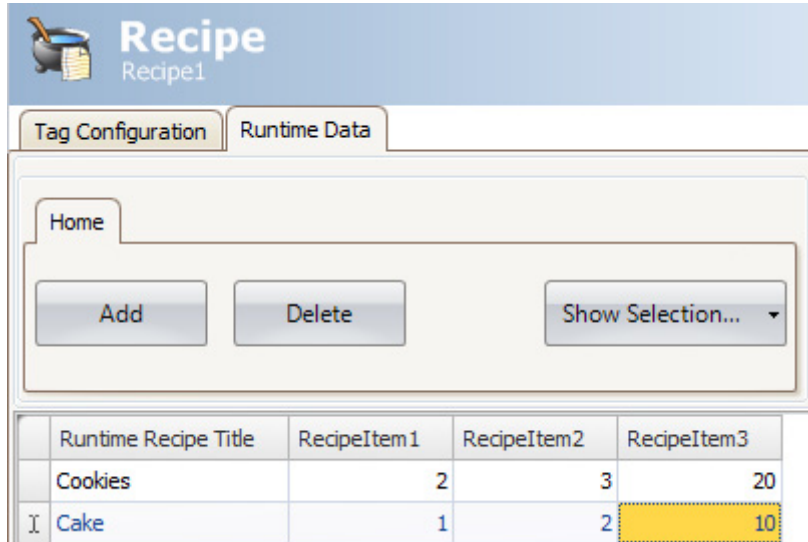
Tag Configuration Tab

First add tags to be used by the recipe function from the tags configuration page. Then define a set of recipe items on the Tag Configuration tab of the Recipe function, and connect them to the tags you defined. The data type is already set in the configuration of the tags.
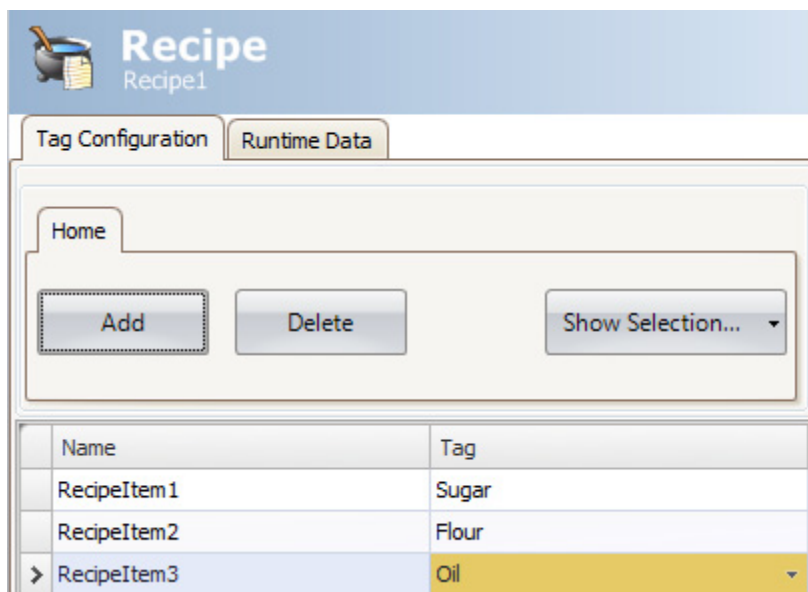
Runtime Data Tab

The actual recipes are created under the Runtime Data tab. Give the recipe a title, and type a value for each recipe item. A recipe can include a maximum of 256 items.
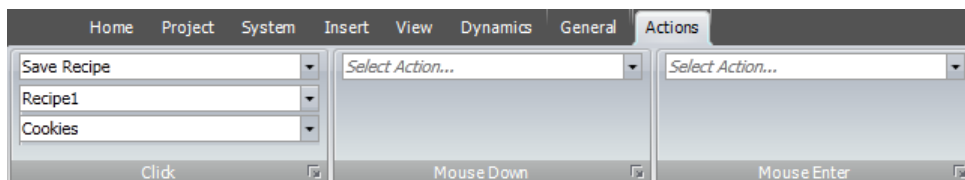
13.2 Recipes in the Operator Panel

Some recipe actions can be performed by the operator in runtime.

13.2.1 Loading Recipes

Recipes can be transferred to the controller using a **Load recipe** action when the operator panel is running. This means that the recipe values are transferred to the controller. A list of available recipes is displayed when the **Load recipe** action is activated. Select a recipe in the list and press [**Enter**] to make the controller run with the downloaded values.

13.2.2 Saving Recipes

The **Save recipe** action can be used when the operator panel is running. Tags defined on the Tag Configuration tab are included in the recipe. Corresponding tag values are stored in the selected recipe file when **Save recipe** is performed. It is possible to define an explicit recipe, that will be overwritten if it already exists when performing the Save Recipe action.

13.2.3 Creating Recipes in the Operator Panel

The Save recipe action is also used to create recipes in runtime. Instead of defining an explicit recipe to overwrite, the recipe data field is left open. In this case, the operator will be prompted to save the recipe with any name, or to overwrite an existing recipe.
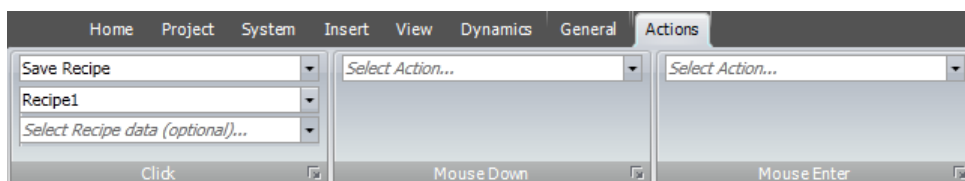
13.2.4 Editing Recipes Offline

It is possible to edit existing recipes in the panel without transferring values to the controller, by setting the recipe tags in offline mode. Recipe tags in offline mode will not receive any data from the controller, and no values will be written to the controller.

In offline mode, the Load recipe and Save recipe actions will not affect values in the controller.

Note:
Internal tags may be used for recipes, but will not be affected by offline mode.

Actions are used to start and end offline mode:

| Action	Description
Start offline recipe editing	Sets the tags in the specified recipe in offline mode. A notification message is shown.
End offline recipe editing	Sets the tags in the specified recipe in online mode, and the tags are updated from the controller.

A notification message is displayed during offline mode.

Example

1. Click **Recipe** on the Insert ribbon tab.
   The Recipe configuration page opens in the desktop area.
2. Add recipe tags on the **Tag Configuration** tab of the Recipe configuration page and connect them to controller tags.
3. Add a recipe on the **Runtime Data** tab.
4. Create a screen with four buttons.
5. Connect two of the buttons to the **Load Recipe** and **Save Recipe** actions. Select the recipe title from the Runtime Data tab for **Recipe Data**.
6. Connect the other two buttons to the **Start offline recipe editing** and **End offline recipe editing** actions.
7. Run the project and test the functionality:
   a. Click **Start offline recipe editing**.
      The tags are set in offline mode.
   b. Click **Start offline recipe editing**.
      The tags are filled with data from the recipe.
   c. Change the data.
   d. Click **Save recipe**.
      The data in the tags is written to the recipe.
   e. Click **End offline recipe editing**.
      The tags are set in online mode and the values are updated with data from the controller.
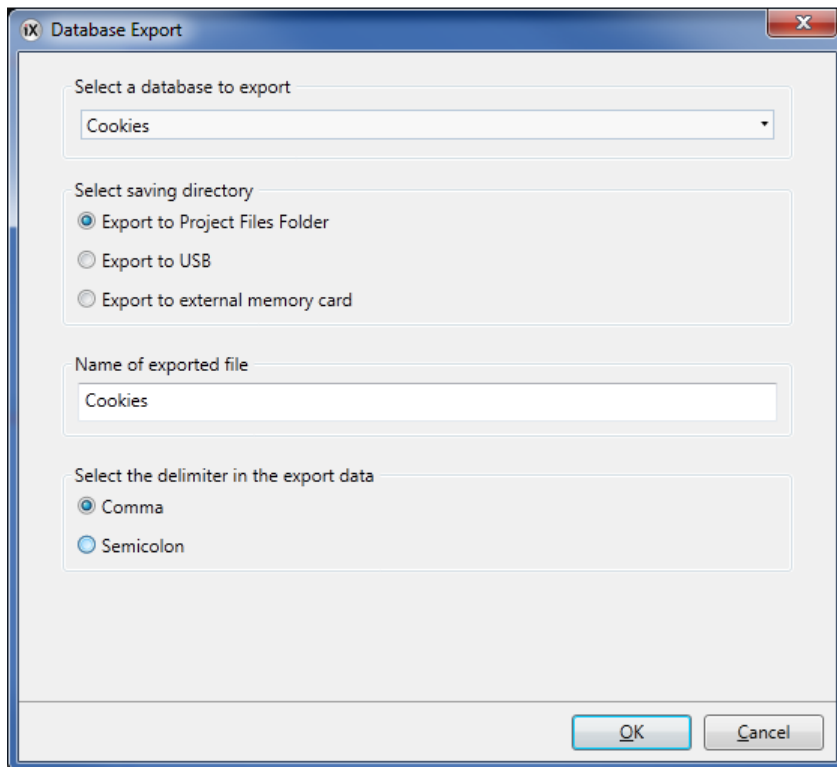
13.3 Recipe Export

A recipe can be exported as a .csv file and saved to a USB stick, an external memory card or to the project files folder. The export is configured through the "**Export Recipe**" action available from the Recipe action group.
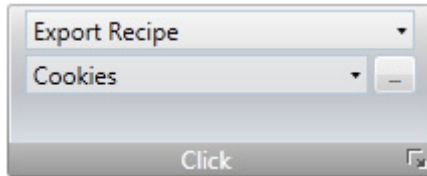
Note:
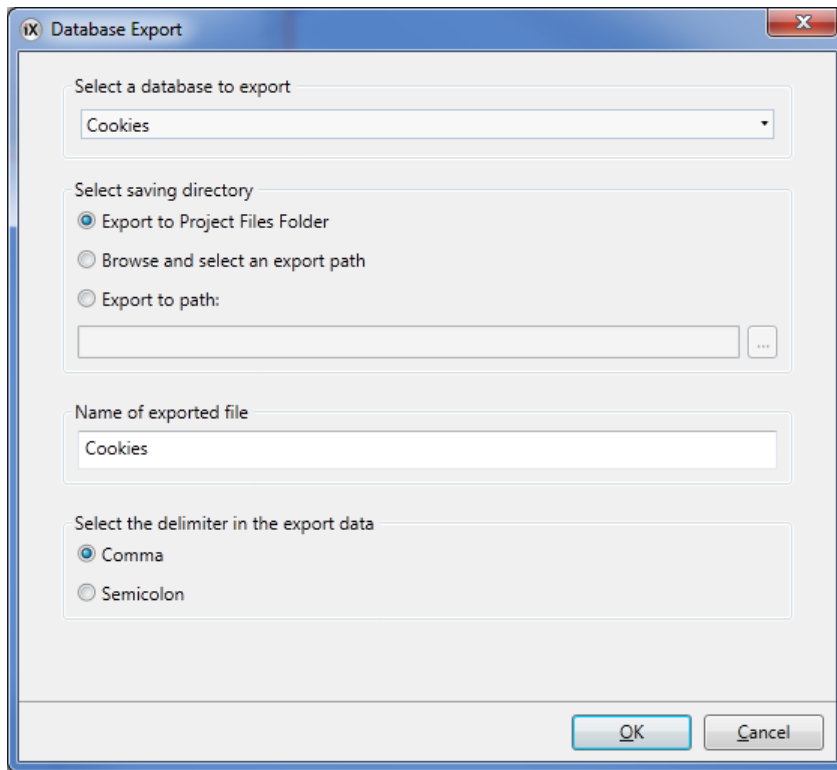Recipe export is not supported during project simulation.

13.3.1 Recipe Export from an Operator Panel Target

It is possible to select **export to USB**, **export to external memory card** or **export to the Project Files folder**, and also setting another name of the export file.

It is also possible to select the requested delimiter in the .csv file (comma or semicolon).

13.3.2   Recipe Export from a PC Target

It is possible to select **Browse and Select an export path**, **Export to Path** or the **Export to Project Files folder**, and also setting another name of the export file.

The **Browse and Select an export path** option allows browsing for desired export destination in runtime. This may be useful if available folders in runtime are unknown when designing the project.

Select the **Export to Path** option when you want to set the path when designing the project. If the pre-defined path is not valid in runtime, the operator may browse the PC environment by clicking a browse button.

It is also possible to select the requested delimiter in the .csv file (comma or semicolon).
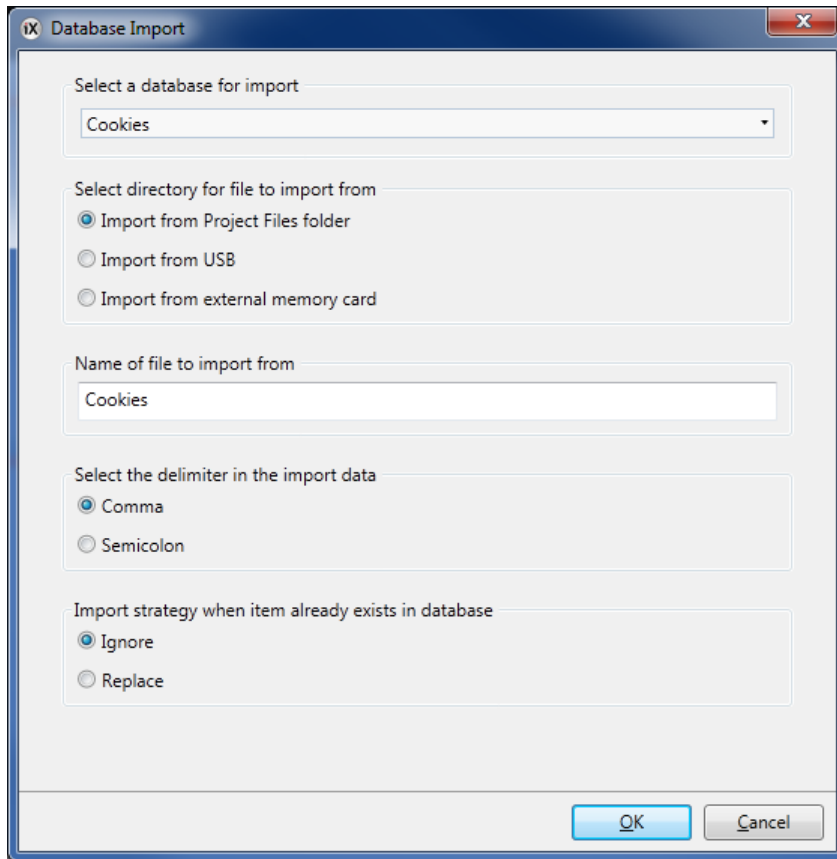
13.4 Recipe Import

A recipe that previously has been exported as a .csv file, can be imported back into recipe database. The import is configured through the "**Import Recipe**" action available from the Recipe Action group.

Import Recipe	▾
Cookies	▾ ...
Click	↘

Note:
Recipe import is not supported during project simulation.

## 13.4.1 Recipe Import to an Operator Panel Target

It is possible to import the csv file using the **Import from USB**, **Import from external memory card** or the **Import from Project Files folder** selections and also setting the name of the import file.

It is also possible to select the requested delimiter in the .csv file (comma or semicolon).

It can be selected whether to ignore the new data or replace the old data, if the recipe data already exists.

13.4.2   Recipe Import to a PC Target
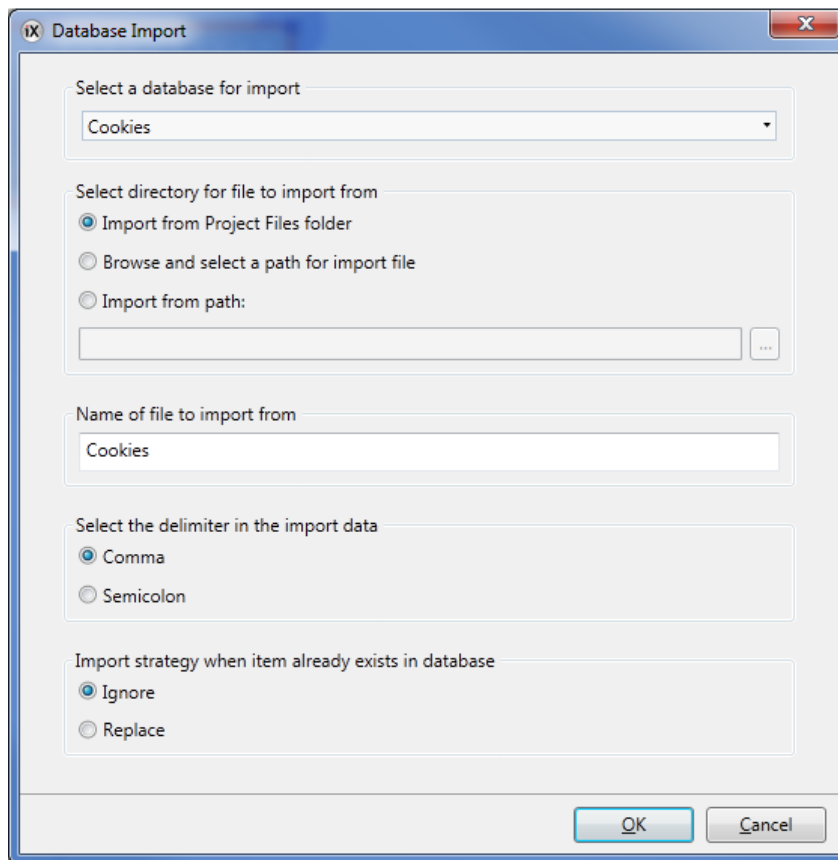
It is possible to select **Browse and Select a path for import file**, **Import from path** or **Import from Project Files folder**, and also setting the name of the import file.

The **Browse and Select a path for import file** option allows browsing for desired import destination in runtime. This may be useful if available folders in runtime are unknown when designing the project.

Select the **Import from path** option when you want to set the path when designing the project.

It is also possible to select the requested delimiter in the .csv file (comma or semicolon).

It can be selected whether to ignore the new data or replace the old data, if the recipes already exists.

14   Function Keys

Function keys can be configured to run actions or to run scripts.

14.1 Definitions

A standard PC keyboard has 12 function keys, F1 - F12.

A function key can be configured to affect only a particular screen or to be global. Global function keys behave identically for all screens. A global definition can always be accessed when the operator panel is running, provided that the current screen does not have a screen definition for that function key. Screen function keys have higher priority than global function keys. Function keys included in background screens have lower priority than screen function keys, but higher priority than global function keys.

Note:
It is not possible to restrict use of function keys based on security groups/login.

The Function keys function is pre-defined in the Project Explorer when a keyboard panel is selected as target, for configuration of the function keys of the selected panel. For touch panel and PC targets, the Function keys function is added via the **Insert** ribbon tab, and allows configuration of up to 24 function keys (even if the connected keyboard does not support more than 12 function keys).

When converting a keyboard panel project to a touch panel or PC project, the function keys will still be included in the project. When converting a touch panel or PC project to a keyboard panel project, you will need to add the function keys from the **Insert** ribbon tab.

Related information
Security Management

14.1.1   Keyboard Operated Panels

The keys of a keyboard operated panel is numbered according to below:
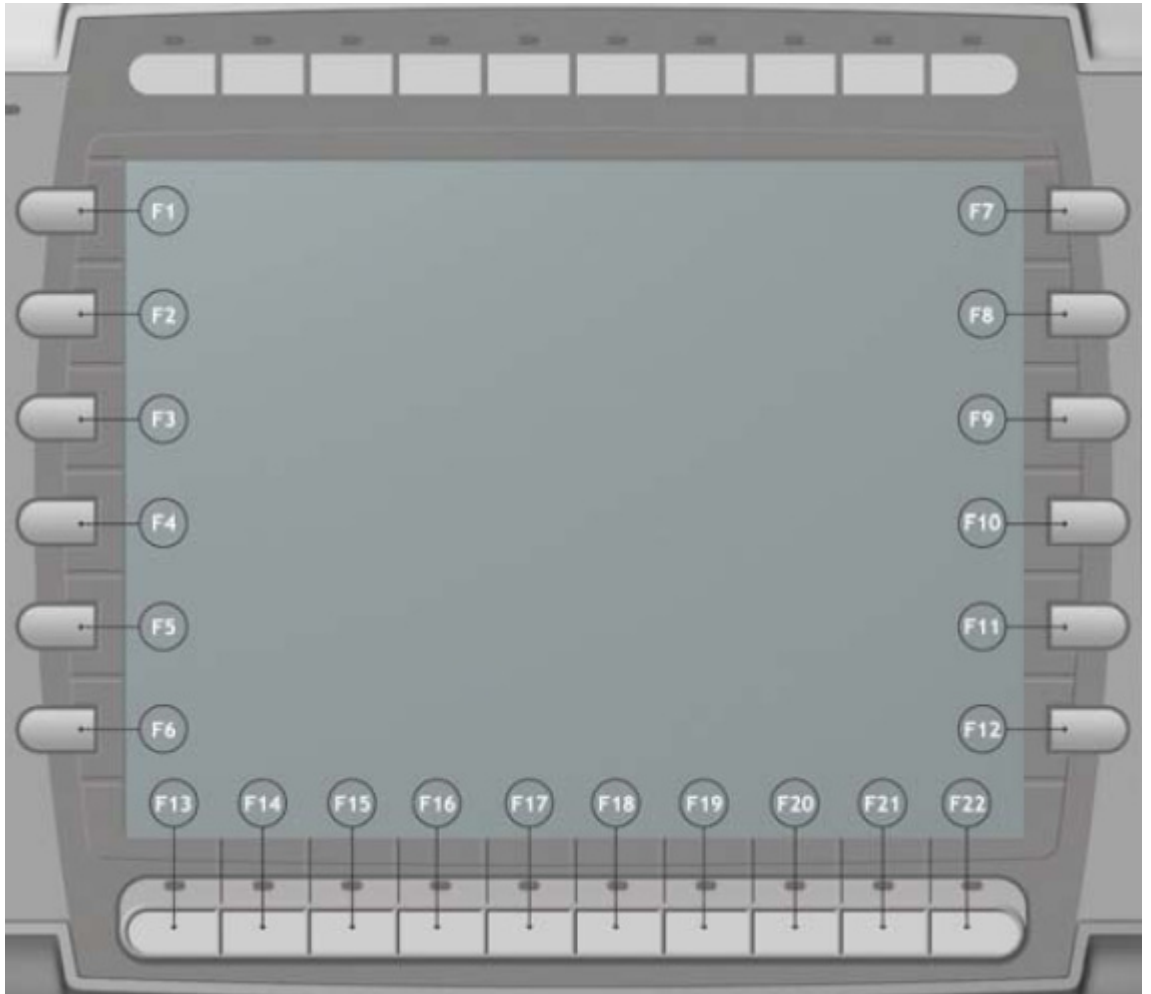
The number of function keys differ between different panel models, but the numbering method is the same: The numbering starts from top downwards on the left side, continues from top downwards on the right side, and finishes from left to right at the bottom row.

The keyboard operated panels also include function keys with pre-defined functions, such as **Prev**, **Next** and **Home**. These are used to navigate between screens in the running project. When dialogs are opened, the **Prev** and **Next** keys are used to navigate between objects in the dialog.

Special functionality for the pre-defined function keys is also used for the Alarm Viewer object in runtime.

| Related information |
|---|
| *Alarm Viewer in Keyboard Operated Panels* |

14.2 Configuring Function Keys

Function keys can be configured to run actions or to run scripts.
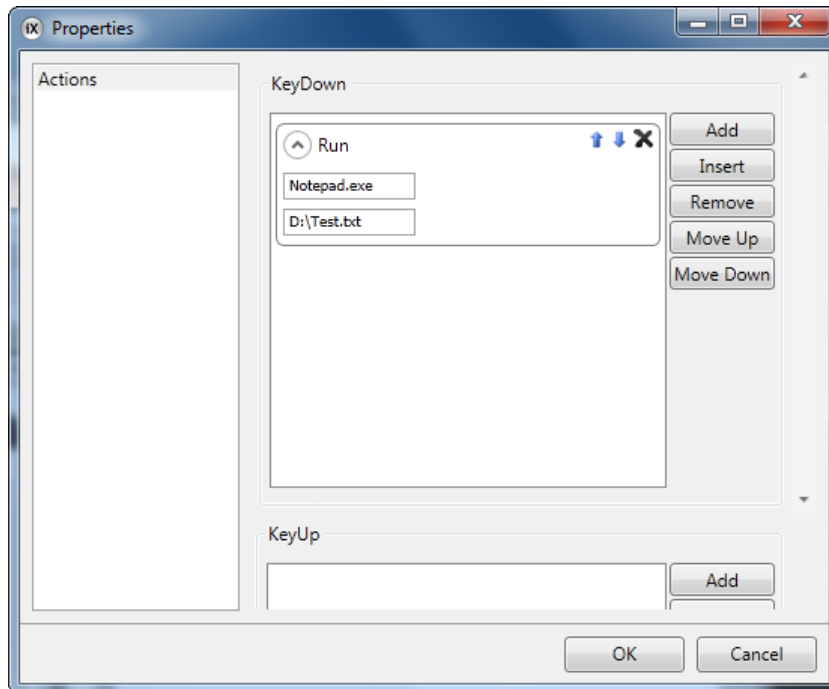
14.2.1 Function Key Actions

To define function key actions:

1. Click on **Function Keys** in the Project Explorer to open the function keys configuration page.

2. Select **Global** or **Screen**.

3. Select the function key to be defined.
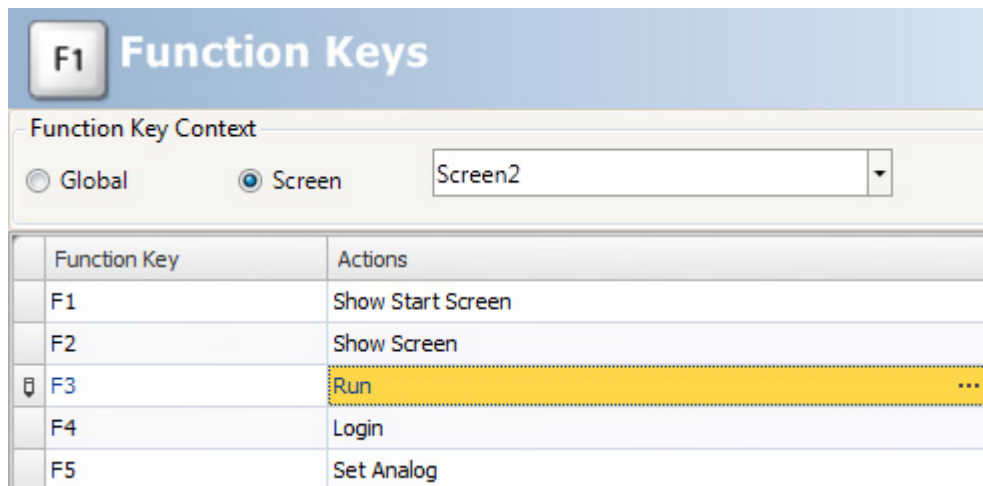
4. Click on the button next to the action to display the actions editor. Define the action to be performed when the function key is pressed or released.

Multiple actions can be configured for a function key.

It is only possible to activate two tags connected to function keys at the same time. This means that if more than two function keys are pressed at the same time, only the first two will be activated.

Note:
The function keys will be disabled if the backlight of the panel is disabled.

Related information
Function Key Action Triggers
Multiple Actions

14.2.2   Function Key Scripts

To define a script for a function key:

1. Click on **Function Keys** in the Project Explorer to open the function keys configuration page.
2. Select the function key to be defined.
3. Select script view mode by clicking **Script** in the lower part of the desktop area.
4. Click on the desired button node. Double-click on the **KeyDown** node. Edit the script code to be performed.

Example:

The following C# code sets up function key F12 to perform an acknowledge of all alarms:

```csharp
public partial class FunctionKeys
{

    void F12_KeyDown(System.Object sender, System.EventArgs e)
    {
        Globals.AlarmServer.Acknowledge();
    }
}
```

Related information
Script

15   Alarm Management

An alarm is used to call attention to an event that requires immediate action. An alarm is set when a certain condition is met. An alarm condition is designed as a logical evaluation of a tag value. Alarms can be divided into groups to create an order of priority.

Alarm notification can be made in the following ways:

Alarm Indicator
Alarm Viewer
Distribution via SMS, e-mail or printer via the *Alarm Distributor*

The alarm server database can be exported in csv format using the **Database Export** action.

Related information
Database Export

15.1 Alarm Conditions

Alarms can assume the following conditions:

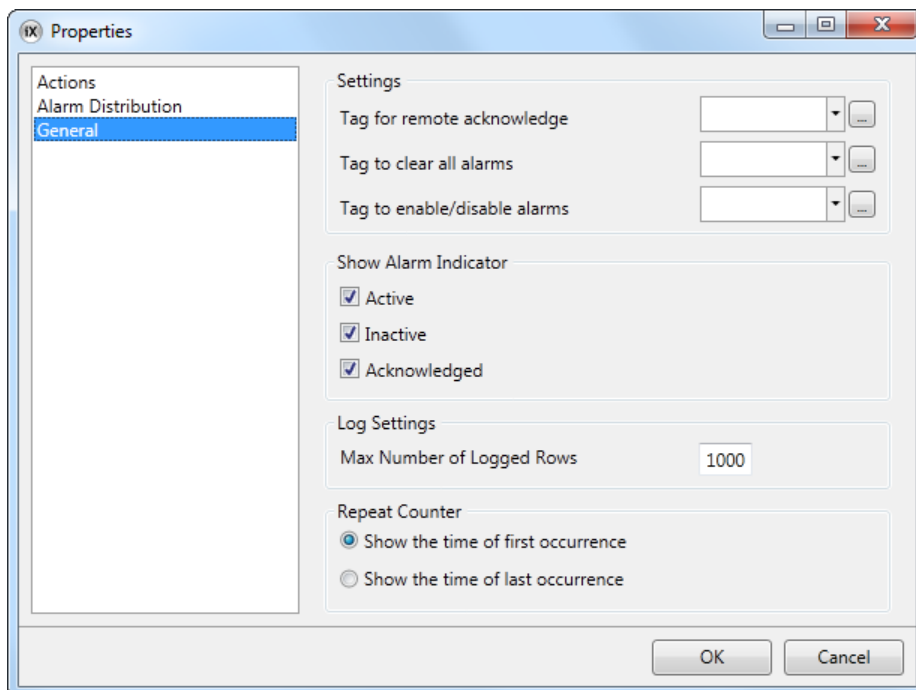| Alarm condition	Description
Active	The alarm condition is met, and the alarm is not acknowledged
Inactive	The alarm has returned to normal condition, but has not been acknowledged
Acknowledged	The alarm condition is met, and the alarm is acknowledged
Normal	The alarm has returned to normal condition, and has not been acknowledged

15.2 Alarm Server

The alarm server function is predefined in the Project Explorer. Clicking on it opens the alarm server configuration pages in the desktop.

General alarm server properties are available by clicking the **Settings** button.
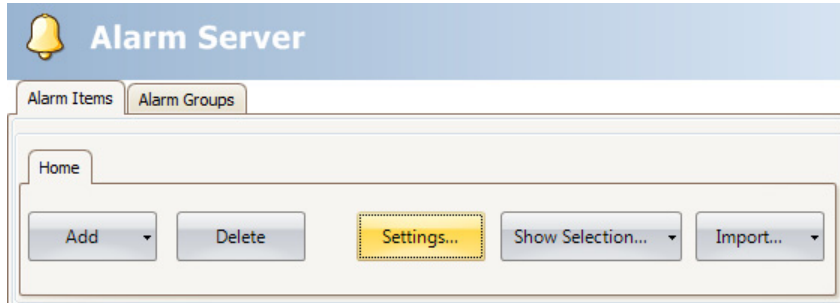
The Properties window is divided into General, Alarm Distribution and Actions.

15.2.1 General Settings

Settings

Some alarm server functions are based on changed tag values:

Alarm function	Description
Remote acknowledge	Performs remote acknowledge of all current alarms when the tag value is 1
Clear alarms	Removes all alarms from the alarm server when the tag value is 1
Enable / disable alarms	Enables the alarm server when the tag value is 1

Show Alarm Indicator

Select for which alarm statuses the alarm indicator is to be displayed.

Log Settings

Parameter	Description
Max Number of Logged Rows	The maximum number of alarms to save in the database. If the number of collected items becomes larger than the set value, the oldest item will be deleted. The value must be within the size of the available amount of memory in the operator panel in runtime.

Repeat Counter

To avoid that an alarm that is triggered repeatedly results in multiple entries in the alarm list, **Repeat Count** can be used. The current alarm will then only appear once in the alarm list, and number of times the alarm is triggered will be included in the alarm text. For display in the Alarm Viewer and for logging, select whether to show the time of the first or last occurrence of the alarm.

15.2.2   Alarm Server Events and Actions

Actions can be configured for the entire alarm server. One or multiple actions can be configured when a particular alarm server event occurs.

Select **Actions** in the Alarm Server properties page to configure actions for the alarm server.

Actions for the following alarm server events can be configured:

Alarm Acknowledge

The specified actions will be triggered once for every alarm that is acknowledged.

The actions will be applied for the individual alarm first. If no action is specified for the alarm, the actions will be applied for the group. If no action is specified for the group, the actions will be applied for the alarm server.

Alarm Active

The specified actions will be triggered once for every alarm that becomes active.

The actions will be applied for the individual alarm first. If no action is specified for the alarm, the actions will be applied for the group. If no action is specified for the group, the actions will be applied for the alarm server.

Alarm Deleted

The specified actions will be triggered once when the **Clear** button in the alarm viewer is pressed, if at least one alarm is deleted.

Alarm Event Info Requested

The specified actions for the selected alarm will be triggered when the **Info** button in the alarm viewer is pressed.

The actions will be applied for the individual alarm first. If no action is specified for the alarm, the actions will be applied for the group. If no action is specified for the group, the actions will be applied for the alarm server.

Alarm Inactive

The specified actions will be triggered once for each alarm, for which the alarm condition is no longer met.

The actions will be applied for the individual alarm first. If no action is specified for the alarm, the actions will be applied for the group. If no action is specified for the group, the actions will be applied for the alarm server.

Alarm Normal

The specified actions will be triggered once for each acknowledged alarm, for which the alarm condition is no longer met.

Alarms Changed

This event is intended primarily for scripting, in order to avoid performance loss that may occur when many alarms change status at once. Alarms Changed includes status changes based on Alarm Acknowledge, Alarm Active, Alarm Normal and Alarm Inactive, but not Alarm Deleted.

The specified actions will be triggered once when the status of several alarms change at one occasion. A collection of alarms is returned.

Alarms Deleted

This event is intended primarily for scripting, in order to avoid performance loss that may occur when many alarms are deleted at once, by pressing the **Clear** button in the alarm viewer.

The specified actions will be triggered once when a number of alarms are deleted at one occasion. A collection of alarms is returned.

Any Acknowledged

The specified actions will be triggered based on alarm logic: when the alarm server toggles between containing acknowledged alarms or not.

Any Active

The specified actions will be triggered based on alarm logic: when the alarm server toggles between containing active alarms or not.

Any Inactive

The specified actions will be triggered based on alarm logic: when the alarm server toggles between containing inactive alarms or not.

15.2.3   Alarm Distribution Settings

To enable sending alarms to printer, or via SMS or e-mail, the Alarm Distributor is used. In addition to the settings below, the **Enable Distribution** option for the alarm groups or alarm items also has to be checked.
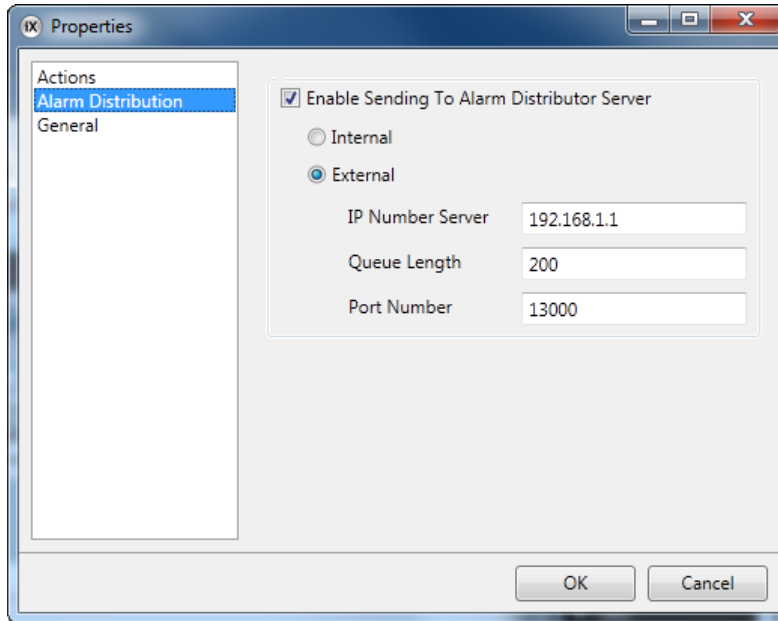
Parameter	Description
Internal / External	Selection of internal or external alarm distributor
IP Number Server	The IP address of the external alarm distributor
Queue Length	The number of alarms to be kept in queue for the external alarm distributor. When the queue is full, no more alarms will be added to the queue.
Port Number	The port number of the external alarm distributor

Related information
Alarm Distributor
Alarm Groups
Alarm Items

15.3 Alarm Indicator

As long as there are active alarms in the alarm server, the alarm indicator notify window will be displayed, regardless of which screen is active. The color shows the current alarm status, and can be configured for each of the alarm groups. The indicator starts in the upper left corner of the screen in runtime, but can be moved to any position on the monitor in runtime.

The appearance of the alarm indicator depends on the current alarm status. It will show the most severe status of the alarms in the list. The Alarm Indicator disappears when all alarms have been acknowledged, and all alarms have returned to inactive status.

Select **General** settings in alarm server properties page to decide for which alarm statuses to show the alarm indicator.

Alarm condition	Description	Default alarm indication
Active	The alarm condition is met, and the alarm is not acknowledged	Flashing red
Inactive	The alarm has returned to normal condition, but has not been acknowledged	Flashing green
Acknowledged	The alarm condition is met, and the alarm is acknowledged	Flashing green
Normal	The alarm has returned to normal condition, and has not been acknowledged	—

Related information
General Settings

15.4 Alarm Items

Alarm items are added from the Alarm Items tab of the alarm server configuration pages.

Click **Add** to define a new alarm.

	Name	Text	Tag	Expression	Condition	Trigger Value	History
	AlarmItem0	SliderMaxValue	MeterAndSlider		GreaterThan	90	☑
>	AlarmItem1	Overflow ···	BoolAlarmTag ▼	···	EqualTo ▼	1	☑

Acknowledge Required	Remote Acknowledge	Remote Ack Expression	Enable Distribution	Repeat Count	Action
☑			☐	☐	
☑	▼	···	☐	☐	Show Screen ···

Parameter	Description
Name	Symbolic name for the alarm item that will show in an alarm viewer.
Text	An optional alarm text that will be shown in the alarm viewer. Parts of the text can be made dynamic by clicking …
Tag	The tag (digital or analog), which generates the alarm when it assumes the specified status.
Expression	See section *Expressions*
Condition	**Equal to/Not equal to**: The alarm is set when the value of the tag is equal/not equal to the value specified in the **Trigger Value** field. **Greater than/Less than**: The alarm is set when the value of the tag is greater/less than the value specified in the **Trigger Value** field. **Equal to greater than/Equal to less than**: The alarm is set when the value of the tag is equal to or greater/less than the value specified in the **Trigger Value** field. **Rising/Falling Edge**: The alarm is set when the specified tag goes to one/zero.
Trigger Value	Numeric value to trigger alarm with when: <tag value> <condition> <trigger value> = true.
History	Specifies when the alarm is to be removed from the alarm list. Checking this box means that the alarm will remain in the list until the list is full. Leaving it unchecked means that the alarm will be removed from the list when it is acknowledged and no longer active. If **Acknowledge Required** is not checked, the alarm will be removed from the list as soon as it is no longer active.

Parameter	Description
Acknowledge Required	Indicates whether the alarm is to be acknowledged or not. Checking the box means that the alarm must be acknowledged, leaving it unchecked means that the alarm returns to normal directly when the alarm status becomes inactive.
Remote Acknowledge	A digital tag that acknowledges all the alarm when set to 1
Remote Ack Expression	See section *Expressions*
Enable Distribution	Enables distribution of the alarm via printer/SMS/e-mail. Only available if alarm distribution is enabled for the alarm server. If the **Enable Distribution** option for the alarm group is checked, distribution of individual alarms in the group are automatically enabled.
Repeat Count	To avoid that an alarm that is triggered repeatedly results in multiple entries in the alarm list, **Repeat Count** can be used. The current alarm will then only appear once in the alarm list, and the number of times the alarm is triggered will be included in the alarm text. It is also possible to show a **Count** column in the Alarm Viewer. The active time can be shown either for the first time or for the last time the alarm occurred.
Action	Makes it possible to configure one or multiple actions when a particular alarm event occurs for the current alarm.

Note:

An alarm condition is triggered by a fixed value. The trigger value defined for an analog alarm tag cannot be controlled from a register. It is not supported to trigger alarms on intervals.

Related information
Actions and Events for Alarm Items and Alarm Groups

15.4.1 Exporting and Importing Alarm Items

Alarm items can be exported and imported using the **Import/Export** button in the Alarms configuration page. The procedure is similar to exporting and importing tags.

Related information
Importing and Exporting Tags
Tag Import Example

15.5 Alarm Groups

Alarms can be divided into groups, for example to indicate the priority of the alarms. One alarm group is defined by default.

Color attributes can be individually set for each alarm group. Alarms can be sorted by group in the alarm viewer.

15.5.1   Defining Alarm Groups

Alarm groups are added from the **Alarm Groups** tab of the alarm server configuration pages.

Parameter	Description
Name	An optional name for the alarm group.
Text	An optional text for the alarm group.
Colors	Select foreground and background color for each alarm status
Remote Acknowledge	A digital tag that acknowledges all the alarms in the group when set to 1
Remote Ack Expression	See section *Expressions*
Enable Distribution	Enables distribution of the alarms in the group via printer/SMS/e-mail. Only available if alarm distribution is enabled for the alarm server.
Action	Makes it possible to configure one or multiple actions when a particular alarm event occurs for the current alarm group.

Related information
Actions and Events for Alarm Items and Alarm Groups

15.6 Actions and Events for Alarm Items and Alarm Groups

Actions can be configured for individual alarms or for alarm groups. One or multiple actions can be configured when a particular alarm event occurs for the current alarm or alarm group.

Actions for individual alarms are configured from the **Action** column for the alarm on the **Alarm Items** tab.

Actions for alarm groups are configured from the **Action** column for the alarm group on the **Alarm Groups** tab.

15.6.1 Actions and Events

Actions for the following alarm events and alarm group events can be configured:

Alarm Acknowledge

The specified actions will be triggered when the selected alarm/an alarm in the selected alarm group is acknowledged.

Alarm Active

The specified actions will be triggered when the selected alarm/an alarm in the selected alarm group becomes active.

Alarm Event Info Requested

The specified actions will be triggered when the Info button in the alarm viewer is pressed.

These actions will be applied for the individual alarm first. If no action is specified for the alarm, the actions will be applied for the group. If no action is specified for the group, the actions will be applied for the alarm server.

Alarm Inactive

The specified actions will be triggered when the alarm/an alarm in the selected alarm group becomes inactive.

15.6.2   Example

The following picture shows two actions configured for when the current alarm goes active, and another action for when the operator presses the **Info** button in runtime.
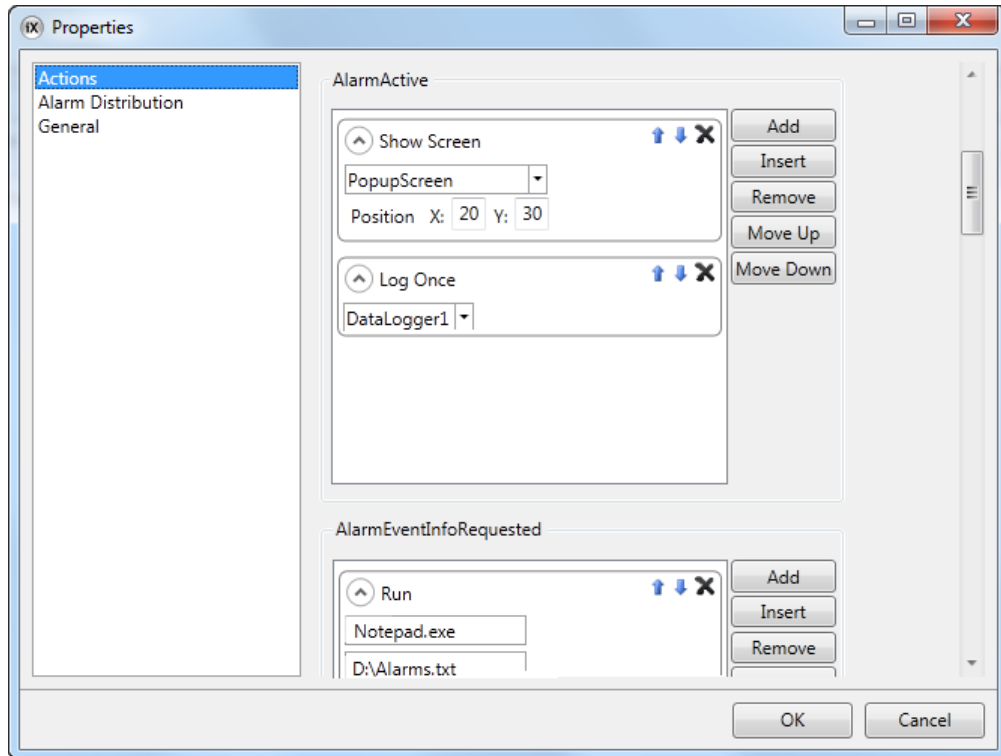
15.7 Remote Alarm Server

The alarm server in the iX Developer project can operate as a stand-alone alarm server, processing alarms and showing these in the alarm viewer and alarm indicator. Alternatively, the alarm server can distribute alarms to clients. Each alarm server can function as alarm server and/or alarm client. The remote alarm server functionality is set up via properties.

## 15.7.1   Remote Alarm Server Limitations

Most functions work exactly the same regardless if a local or remote alarm server is used, but there are a few limitations:

- In projects where multiple languages are used, switching language in the alarm server results in using the current language for all new alarm texts. Alarms distributed before the language was switched in the server will remain in the previous language in the client.
- Pressing the Info button in the alarm viewer in runtime will not result in any actions for a remote alarm, unless the same project is used for the server and the client. The script event AlarmInfoRequested can be used instead.
- Unless the same project is used for the server and the client, filtering (pressing the Filter button in the alarm viewer) can only be made in runtime for remote alarms, since the remote server may include e.g. alarm groups that are different from the project in which the alarm viewer was configured.

15.7.2   Remote Alarm Server Properties

The following properties, located in the **Misc** group, need to be set up for the remote alarm server:

| Property	Description
MaximumAlarm EventsToDistribute	The number of events to distribute to the clients. The number of distributed alarms affects performance.
ServerMode	The alarm server can assume the following modes: **Disabled**: The alarm server is disabled and no alarms are processed **Local**: The alarm server processes alarms of its own. **Remote**: Alarms are distributed to remote clients. **Both**: The alarm server processes alarms of its own and also distributes them to remote clients
ServerPort	Any available port. The port needs to correspond to the port setting for the remote alarm client.
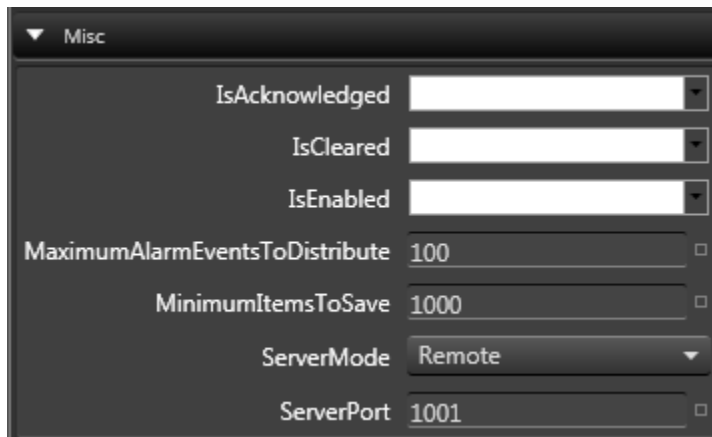
Related information
Property Grid

15.7.3 Remote Alarm Client

The following properties, located in the **Misc** group, need to be set up for the remote alarm client:

Property	Description
ServerAddress	The IP address of the remote alarm server.
ServerPort	Any available port. The port needs to correspond to the port setting for the remote alarm server.
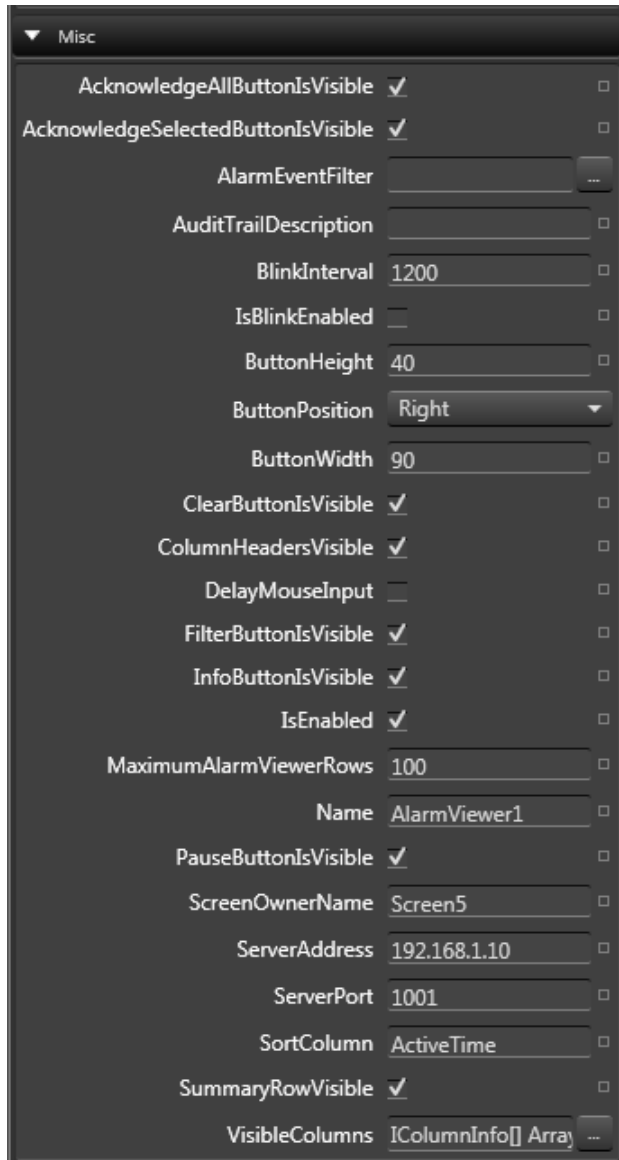
Related information
Property Grid

15.8 Alarm Distributor

The Alarm Distributor makes it possible to send alarm notification via printer, SMS or e-mail. The function can be enabled internally in a project, or in another operator panel that acts as a server towards several connected clients. Regardless of acting as server or client, distributed alarms are saved temporarily in the local project database to ensure that information is not lost in case of e.g. interrupted power. After receiving alarms from a client, the server will send a confirmation to the client. The alarms will then be removed from the client's database.

15.8.1 Alarm Distribution Roles

Using Internal Alarm Distribution Server Functions (Acting as Client and Server)

The following actions are required for using distributed alarm functions in the current project:

Action	Component	Location
Activating distribution of alarms in the project	Alarm Distribution Server	**Insert** ribbon tab
Enabling internal distribution of alarms	Alarm Server	**Settings** button/**Alarm Distribution** properties
Enabling distribution of alarm groups or individual alarms	Alarm Server	**Alarm Groups** or **Alarm Items** tab
Configuration of alarm distribution routes	Alarm Distribution Server	**Receivers**, **Alarm Filter** and **Scheduler** tabs
Configuration of alarm distribution devices	Alarm Distribution Server	**Configure Distribution Devices** button (e-mail, printer and/or SMS)

Using External Alarm Distribution Server Functions (Acting as Client)

The following actions are required for using distributed alarm functions in another operator panel/PC project:

Action	Component	Location
Activating distribution of alarms in the project	Alarm Distribution Server	**Insert** ribbon tab
Enabling external distribution of alarms	Alarm Server	**Settings** button/**Alarm Distribution** properties
Enabling distribution of alarm groups or individual alarms	Alarm Server	**Alarm Groups** or **Alarm Items** tab

Collecting Alarms for Distribution (Acting as Server)

The following actions are required for using distributed alarm functions in another operator panel/PC project:

Action	Component	Location
Activating distribution of alarms in the project	Alarm Distribution Server	**Insert** ribbon tab
Configuration of alarm distribution routes	Alarm Distribution Server	**Receivers**, **Alarm Filter** and **Scheduler** tabs
Configuration of alarm distribution devices	Alarm Distribution Server	**Configure Distribution Devices** button (e-mail, TCP server, printer and/or SMS)

15.8.2   Adding an Alarm Distributor

The Alarm Distributor can be added to the project from the Insert ribbon tab. The configuration pages are opened in the desktop area.

Route

A route defines a number of filtering rules for distribution of alarms. Multiple routes makes it possible to send different kinds of notifications to various receivers.
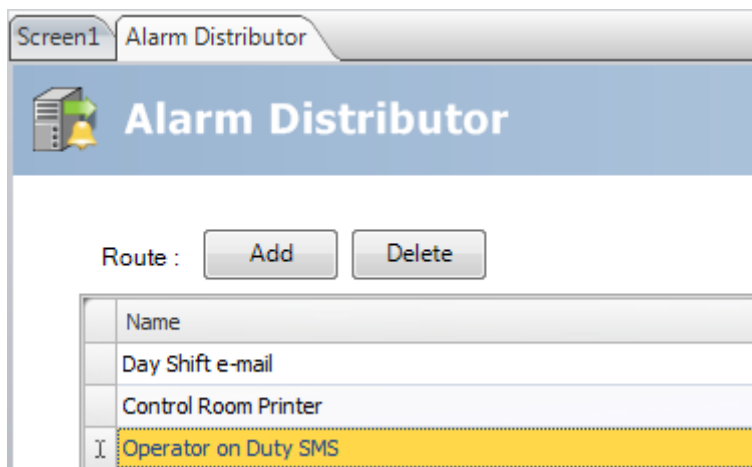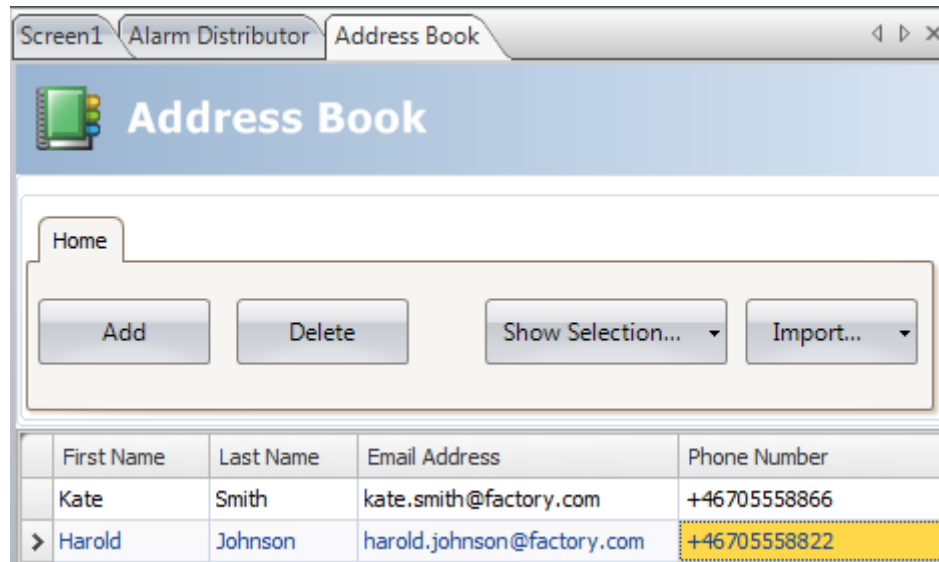
Receivers Tab

The receivers of distributed alarms for each route are set up on the **Receivers** tab.

Address Book

Click on the **Address Book** button to enter names, e-mail addresses and/or phone numbers of the persons that are to be available for selection for distributed alarm messages.

The **Show Selection/Show All** button can be used to set up a filter of displayed items.

The contents of the address book can be exported as a .csv file, and then be imported again after modification using the **Import/Export** button.

Related information
Filtering Items

Receiver Selection

Click on the **Receiver Selection** button to set up receivers of each alarm distribution route; via e-mail, SMS and/or printer.
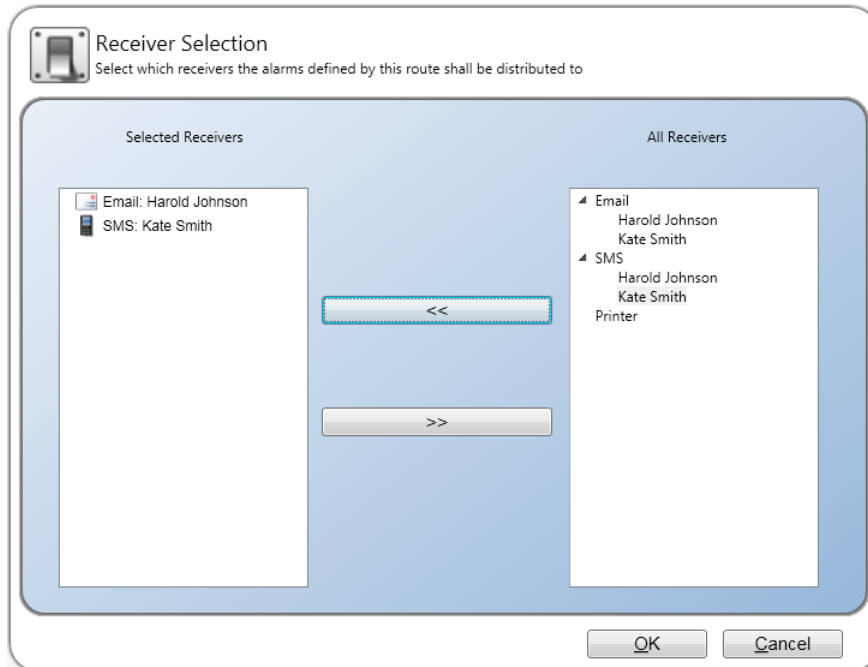
The selected receivers are displayed under **Parallel Receivers**.
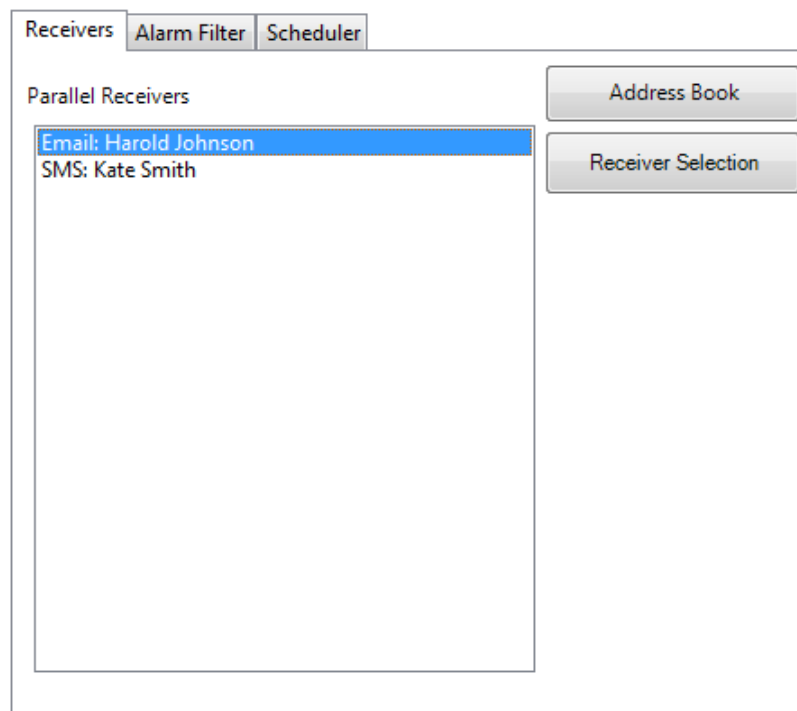
Alarm Filter Tab

Filtering of each alarm distribution route can be configured on the **Alarm Filter** tab. The filtering operates using "and" logic, i.e. the alarm will only be distributed if all conditions are fulfilled. An empty string does not generate a filter condition (acts as a "wild card").

Parameter	Description
Alarm Name	The alarm item to distribute.
Alarm Group	The alarm group of which alarm items are to be distributed.
Alarm Text	All alarms including this text will be distributed.
Alarm Status(es) to distribute	Selection of alarm to be distributed, based on alarm status. At least one status must be selected.

Scheduler Tab

Scheduling of each alarm distribution route can be configured on the Scheduler tab. Only alarms that occur between the start and stop times will be distributed.

	Period	Day	Start Time	Stop Time	Name
I	Weekly	Tuesday	13:40:00	13:50:00	Tuesday 13:40:00 to 1...

Parameter	Description
Period	Select daily or weekly alarm distribution
Day	Select weekday, if **Weekly** was selected for **Period**
Start Time / Stop Time	Select start and stop time of the alarm distribution
Name	The name is automatically filled in based on start and stop time, and cannot be edited

15.8.3   Configure Distribution Devices

Click the **Configure Distribution Devices** button to set up the different devices.

Related information
Modifying Alarm Distribution Settings in Runtime

Alarm Variables

Variables collected from the alarm server definitions can be included in distributed alarm information. Variables are enclosed by curly brackets; for example "Alarm text: {3}". The following variables can be used:

Number	Variable	Description
0	State	The current state of the alarm
1	StateTime	The time the alarm entered the current state
2	AlarmId	A unique alarm ID
3	AlarmText	Alarm item text
4	AlarmItemDisplay-Name	Alarm item name
5	Count	The number of times the alarm has occurred
6	ActiveTime	The time the alarm became active
7	InactiveTime	The time the alarm became inactive
8	NormalTime	The time the alarm became normal
9	AcknowledgeTime	The time the alarm was acknowledged

E-mail Device

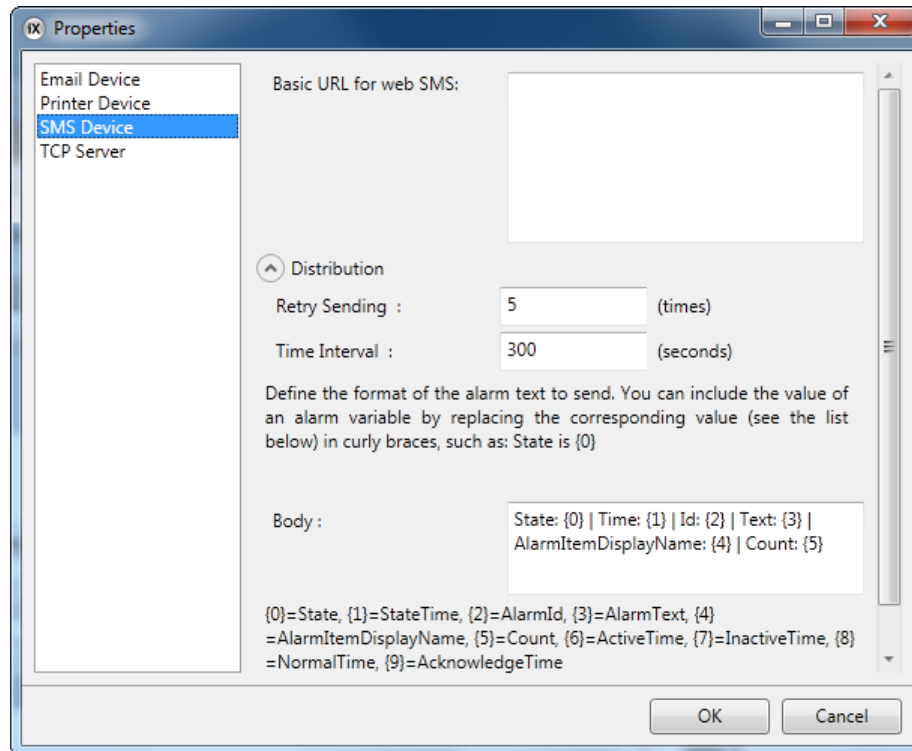| Parameter	Description
From Name	The name of the sender of the e-mail
From E-mail Address	The e-mail address of the sender of the e-mail
SMTP Server Name/IP	The name or IP address of the SMTP server for sending e-mail
Port	The port number of the e-mail server
Code Page	The code page consists of a table of values that describes the character set for a particular language. The default code page in iX Developer is 65001 (UTF-8).
Authentication Mode	Select **None** or **Authenticated Login**
User Name/Password	User name and password if **Authenticated Login** was selected
Retry Sending	Number of retries if the e-mail cannot be delivered
Time Interval	Number of seconds between retries
Subject	The e-mail subject; *Alarm Variables* can be used
Body	The e-mail body; *Alarm Variables* can be used

Printer Device
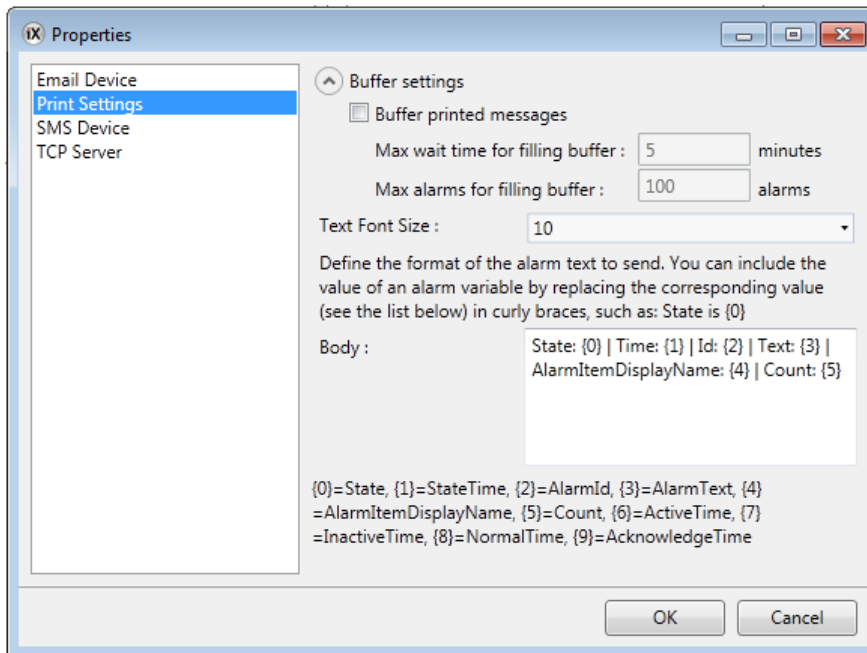
Parameter	Description
Buffer settings	Set up the printers buffer settings
Text Font Size	The font size of the text to be printed
Body	The body of the printout; *Alarm Variables* can be used

Related information
Output Devices Group

SMS Device

To be able to send alarms via SMS, you have to purchase an SMS Gateway Service.

| Parameter	Description
Basic URL for web SMS	The URL configuration string according to the SMS Gateway Service supplier that must include two variables: {0} for telephone number and {1} for SMS body
Retry Sending	Number of retries if the SMS cannot be delivered
Time Interval	Number of seconds between retries
Body	The SMS body; *Alarm Variables* can be used

TCP Server

Define the TCP server port. This must correspond to the port defined for the **Alarm Distribution** settings for the alarm server.

15.9 Alarm Distributor Viewer

The Alarm Distributor Viewer is a debug tool, used for test and verification of the alarm distribution setup.

Clicking the **Settings** button in runtime makes it possible to edit the address book and make settings for alarm distribution.
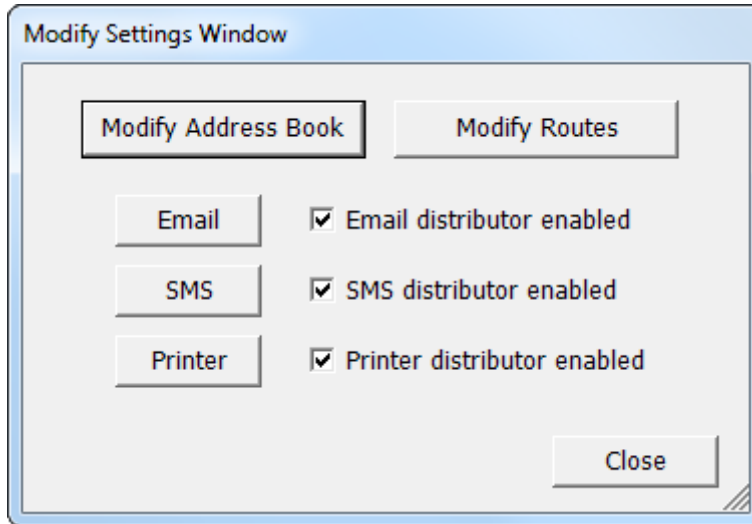
Related information
Modifying Alarm Distribution Settings in Runtime

15.10 Alarm Viewer

The alarm viewer can be used to show more detailed alarm information than the alarm indicator. Alarms presented in the Alarm Viewer presents alarm texts from the alarm server. The alarm list shows the latest alarms. It is arranged in alarm group order.

The default design of the alarm viewer depends on the size of the selected operator panel target, in order to allow full visibility and usability of the alarm viewer in the smaller operator panels.

Settings such as font and button width are available in the Property grid.

Settings for the alarm viewer are available from the **General** ribbon tab while the object is selected:

15.10.1   Buttons Group

Parameter	Description
Position	Controls to which of the borders the buttons will line up
Acknowledge Selected	Acknowledges currently selected alarms in the alarm viewer
Acknowledge All	General acknowledge of all current alarms in the alarm viewer
Clear	Removes all alarms in the viewer
Filter	Allows filtering of alarms in the viewer
Info	Performs the action set by the AlarmEventInfoRequested action
Play/Pause	Starts and pauses the alarm viewer
Enable Context Menu	Show alarm viewer commands in context menu in runtime

15.10.2   Display Settings Group

Parameter	Description
Configure Columns	Selection and configuration of columns to display in runtime
Filter Settings	Filtering of alarm statuses to display in runtime
Max Number of Rows	The maximum number of alarms to display. If the number of alarms does not fit in the object in runtime, scroll bars will be made available. If the number of alarms exceeds the maximum number, the oldest item will be deleted.
Show Column Headers	Shows/hides the alarm viewers column headers
Show Summary	Shows status information in the bottom of the alarm viewer

15.10.3   Alarm Viewer Commands

In addition to using the alarm viewer buttons, the same functions are available using script.

Related information
Script
Property Grid

15.11 Alarm Management in Runtime

Alarm notification can be made in the following ways:

Alarm Indicator
Alarm Viewer
Distribution via SMS, e-mail or printer via the *Alarm Distributor*

15.11.1 Alarm Acknowledgement

An alarm can be acknowledged in the following ways:

- right-clicking on the alarm line and selecting **Acknowledge**
- selecting the alarm line and pressing on the **Acknowledge Selected** button
- pressing the **Acknowledge All** button to acknowledge all current alarms
- script actions

Inactive alarms that have been acknowledged can be removed from the list by pressing the **Clear** button.

15.11.2 Sorting

Pressing on the column heading in runtime will sort the alarm list by that column. This will pause the alarm viewer.

15.11.3 Filtering

Pressing the Filter button in the alarm viewer opens a dialog that allows the user to enter a filtering profile to control which alarms to display in the alarm list.

Filtering can be made based on any combination of the following conditions:

- Alarm group: The first column in the filter window shows the alarm group name. Check the Enable box to show alarms from the group.
- Alarm status: Displays alarms with the status corresponding to the checked status values.

15.11.4 Play/Pause Button

The alarm viewer window can be paused from being updated with incoming alarms using the Play/Pause button. This allows the operator to inspect and sort the list without interference from any incoming alarms. The list will also pause when an alarm is selected.

The alarm list will start updating again when

- the Play button is pressed
- the operator changes screen and returns to the same screen (or goes to another screen showing the alarm viewer)
- after a 5 minute time-out

15.11.5 Info Button

Pressing the Info button in the alarm viewer will trigger the action specified in the **AlarmEventInfoRequested** group on the **Actions** tab, when an item in the alarm server is selected.

This action will be applied for the individual alarm first. If no action is specified for the alarm, the action will be applied for the group. If no action is specified for the group, the action will be applied for the alarm server.

Related information
Alarm Groups
Alarm Conditions
Alarm Event Info Requested

15.11.6 Alarm Viewer in Keyboard Operated Panels

Some of the built-in arrow keys and special keys (such as ACK and Enter) in the keyboard operated panels have special functions in runtime when the Alarm Viewer object is included in the current screen.

Key	Description
Enter	Pressing **Enter** when the cursor is outside of the Alarm Viewer moves the cursor inside the Alarm Viewer.
Backspace	Pressing **Backspace** when the cursor is inside the Alarm Viewer moves the cursor to outside of the Alarm Viewer.
ACK	Pressing **ACK** when the cursor is outside of or inside the Alarm Viewer acknowledges all alarms.
Left arrow	Pressing the **Left** arrow when the cursor is inside the Alarm Viewer puts focus on the alarm list.
Right arrow	Pressing the **Right** arrow when the cursor is inside the Alarm Viewer puts focus on the Alarm Viewer buttons.

Related information
Keyboard Operated Panels

15.11.7 Modifying Alarm Distribution Settings in Runtime

It is possible to configure actions, or to use the Alarm Distribution Viewer to modify the address book and alarm distribution settings in runtime.

Note:
If the Alarm Distribution Viewer or alarm distribution actions are used in runtime, it is possible that alarm distribution settings have been modified. Next time you download a project to the operator panel, you will get a notification, and selection to overwrite panel settings or not.

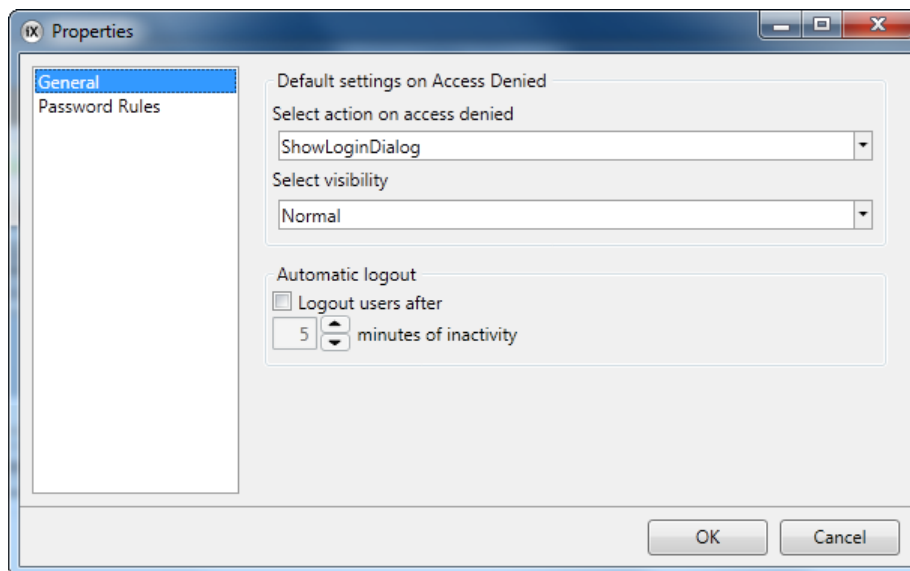| **Related information** |
|---|
| Address Book *Action Groups* |
| Output Devices *Action Groups* |

16   Security Management

Using security groups and passwords in the operator panel makes it possible to create a security system for the project. Operators can easily be assigned different authorizations in the project to restrict access to objects and functions. Security group information and passwords are stored in the project database.

It is not necessary to use this function if unrestricted access is acceptable. The security function is a predefined component of the Project Explorer.

16.1 General Security Settings

General settings for the security function are available by clicking on **Security** in the Project Explorer, selecting the **Users** tab and clicking the **Settings** button.

Parameter	Description
Select action on access denied	When a users's access rights are not sufficient to e.g. modify an object or carry out an action, it is possible to display an access denied message or opening a login dialog. Selecting **None** performs no action.
Select visibility	When a user with insufficient access rights tries to affect an object, it is possible to specify visibility for those objects: **Disabled:** Only users with sufficient access rights can affect the object. The object is visible, but disabled, for users with insufficient access rights. **Hidden:** Only users with sufficient access rights can see the object. The object is invisible for users with insufficient access rights **Normal:** Only users with sufficient access rights can affect the object. The object looks normal for users with insufficient access rights

Parameter	Description
Automatic logout	Checking the box makes it possible to log out users automatically after a certain number of minutes of inactivity. After an automatic logout is executed, a screen jump to the start screen will be performed.

Note:
The automatic logout will not be carried out if a communication error (COM error) is active in the panel. After removing the communication error from the panel, the automatic logout will be performed within the specified time, calculated from when the communication was reestablished.

16.1.1   Password Rules Properties

Allows setting a minimum number of characters for passwords.

Related information
Logging In and Logging Out

16.2 Security Groups

Security groups are defined on the **Groups** tab of the security configuration page. When objects and screens have been set up with restricted access, the current user must belong to a specific security group in order to control or see the object or screen.

16.2.1   Creating Security Groups

A new security group in a project is defined by clicking on **Security** in the Project Explorer, selecting the **Groups** tab and clicking **Add**.

Parameter	Description
Name	Any alphanumeric string, beginning with a letter (a-z, A-Z)
Users	Selection of users defined on the **Users** tab to be included in the group
Users Invisible in Runtime	When the box is checked, the users in the current group will not be available for selection from the **Login** dialog in runtime. It will still be possible to type in the user name and password to login. Operator Panels targets only: In order to enable the possibility to manually enter a user name and password, it is necessary to click the "Login" header on the Login dialog in runtime.

16.3 Users

Users and passwords for login are defined on the **Users** tab of the security configuration page.
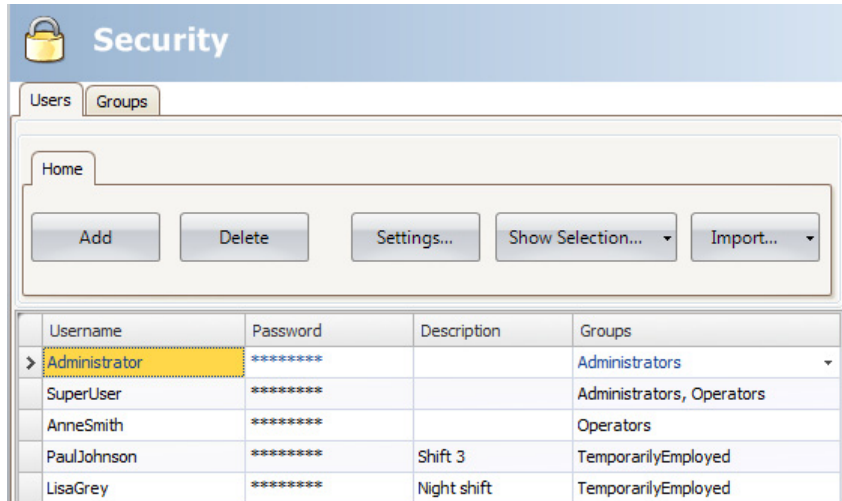
Parameter	Description
Name	Any alphanumeric string, beginning with a letter (a-z, A-Z)
Password	Any alphanumeric string; minimum length according to **Password Rules** properties. The password is converted to asterisks as you leave the password input cell.
Description	An optional description of the user
Groups	Selection of security groups for the user

It is also possible to add users and change passwords directly in a running project, using the **Show Users Dialog** action.

16.3.1   Logging In and Logging Out

The login dialog can be displayed in runtime by, for example, clicking a button, for which the **Login** action has been specified. It is possible to change password from the login dialog.
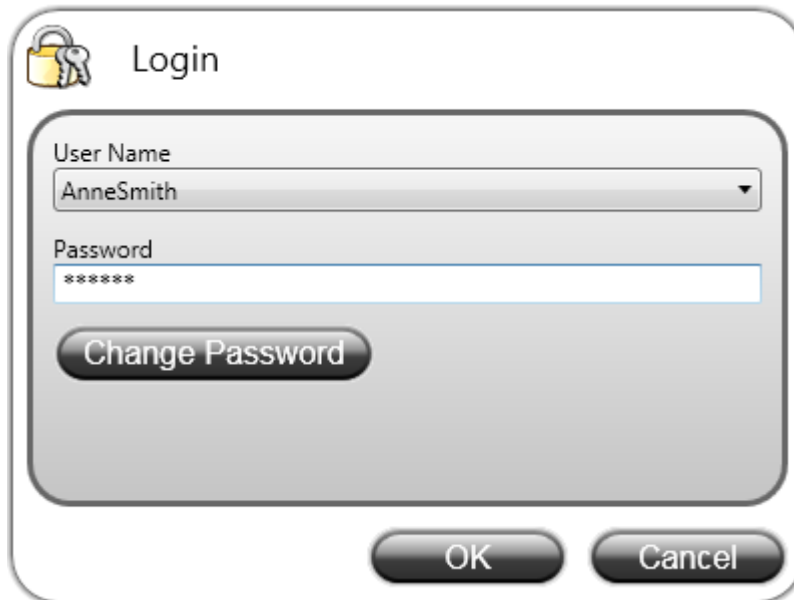
The login dialog lists all users that have been configured for the current project, except for users in groups for which **Users Invisible in Runtime** was selected.

It is also possible to configure a general behavior that brings up the login dialog any time a user tries to affect an object that is not within the current user's access rights. This behavior is configured from the Properties window for the Security function, and selecting **ShowLoginDialog** for action on access denied.

The **Logout** action is used to perform a logout.

Users may also be logged out automatically after a certain number of minutes of inactivity by configuring **Automatic logout** in the **Properties** window for the Security function.

Note:
After logging out, the function **Show Previous Screen** for function keys and touch keys will be ignored. This is a security feature to prevent unauthorized persons from accessing screens protected by passwords.

Related information
General Security Settings
Actions Ribbon Tab

16.4 Object Security and Visibility

Object access can be restricted to a certain security group. Each security group contains a set of users and each user has an individual password. To be able to control an object with a security setting, the user must belong to the defined security group, and has to login.

Object security is controlled by selecting the object, and then opening the Tag/Security group of the Home or General ribbon tabs.

When no security is set for an object, the object will be available to everyone, i.e. login will not be requested.

Note:
It is not possible to set security on function keys.

The visibility and behavior settings specified in the security manager will be used when a security group is set but no explicit visibility is specified. The following options can be set:

Visibility	Description
Default	The object looks normal and is visible for anyone.
Disabled	Only users in the selected security groups can affect the object. The object is visible, but disabled, for users with insufficient access rights.
Hidden	Only users in the selected security groups can see the object. The object is invisible for users with insufficient access rights.
Normal	Only users in the selected security groups can affect the object. The object looks normal for users with insufficient access rights.

Related information
Tag/Security Group
Function Keys
Screen Security

17 Language Management

iX Developer supports multilingual applications. Translation of texts and system texts can be performed directly in the application, or via export to a text file to be translated in other software. The file is imported to the application after translation.

The application language can be changed in runtime, for example based on a tag value.

17.1 Setting Up Multiple Languages

The multiple languages function is included in the application by default and is located in the Functions folder in the Project Explorer.

The multiple languages configuration pages consist of the Texts, System Texts and Languages tabs.

17.1.1 Adding Languages

1. Click on the **Multiple Languages** function to open its configuration pages.
2. Select the **Languages** tab.
3. Add languages by selecting them from the drop-down list.
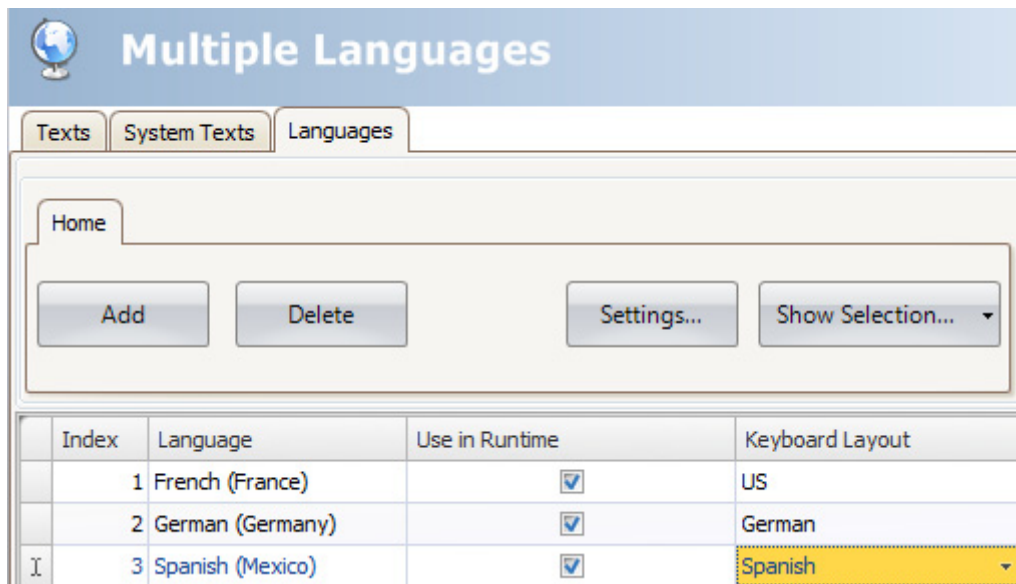
The **Use in Runtime** setting makes the language available in runtime. Under **Keyboard Layout**, it is possible to select different layouts for the virtual keyboard, used for alphanumerical input in runtime.

Each added language is assigned an index number. Index number 0 is reserved for the default language (the language that the application was first created in).

The **Add** and **Delete** buttons can be used to add and delete languages.

The **Settings** button allows connecting the language to a tag. This makes it possible to change the project language based on a changed value.
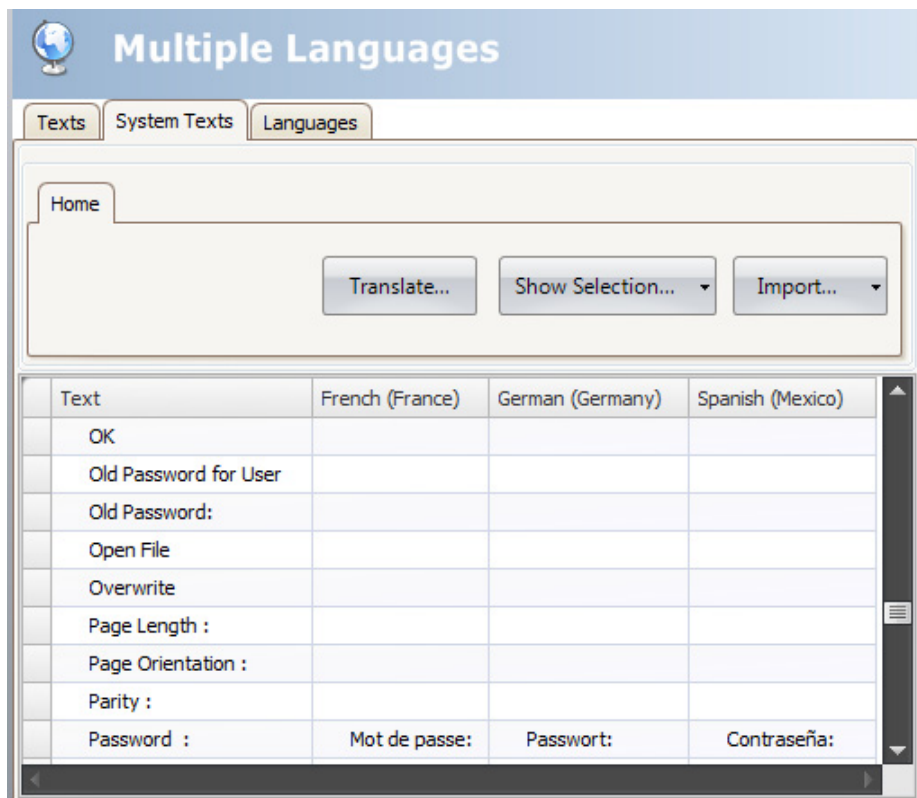
Use the **Show Selection/Show All** button if you want to filter the languages in the current view.

Related information
Virtual Keyboard
Filtering Items

17.2 System Texts

System texts belong to functions (e.g. alarm server or security), to the system (e.g. time zone) and to objects (e.g. buttons or sliders), excluding text that the developer has entered.

Select the **System Texts** tab in the Multiple Languages configuration pages to edit system texts.

Use the **Show Selection/Show All** button if you want to filter the current view.

The **Import/Export** button can be used to export a system language to a text file for translation, and then import the language after translation.

Related information
Filtering Items
Exporting Languages
Importing Languages

17.3 User Texts

Texts that have been added to objects by the developer are available from the **Texts** tab in the Multiple Languages configuration pages. The texts in the default language can be edited as well.
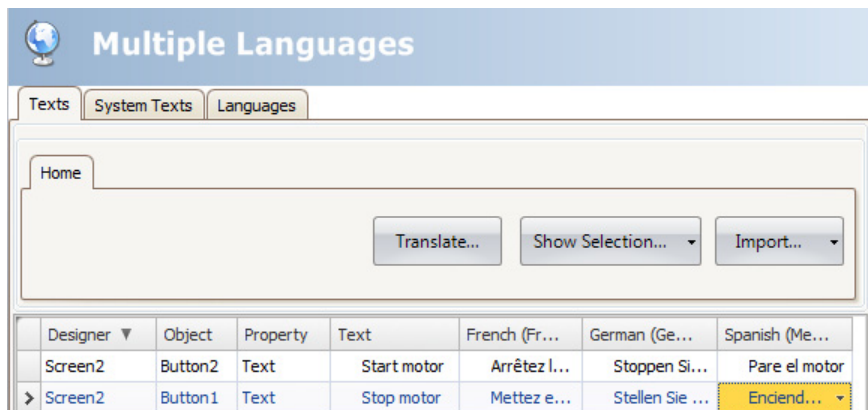
Use the **Show Selection/Show All** button if you want to filter the current view.

The **Import/Export** button can be used to export a system language to a text file for translation, and then import the language after translation.

Note:
When an object is connected to a text in the Text library, the object's internal denomination (that is not displayed in runtime) will be presented in grey color in the Texts list, to indicate that editing the text will not affect the runtime project.

| Related information |
| --- |
| *Filtering Items* |
| *Exporting Languages* |
| *Importing Languages* |
| *Text Library* |

17.4 Automatic Translation

iX Developer includes a translation tool that uses Microsoft Bing Translator to offer quick translations of user-defined or system texts free of charge.

Note:
Automatic translations provide limited functionality; especially combination of words or words with synonyms may be returned as incorrect translations.

The translation tool is available by clicking **Translate** on the Texts or System Texts tabs of the Multiple Languages configuration pages. Each language is translated separately.
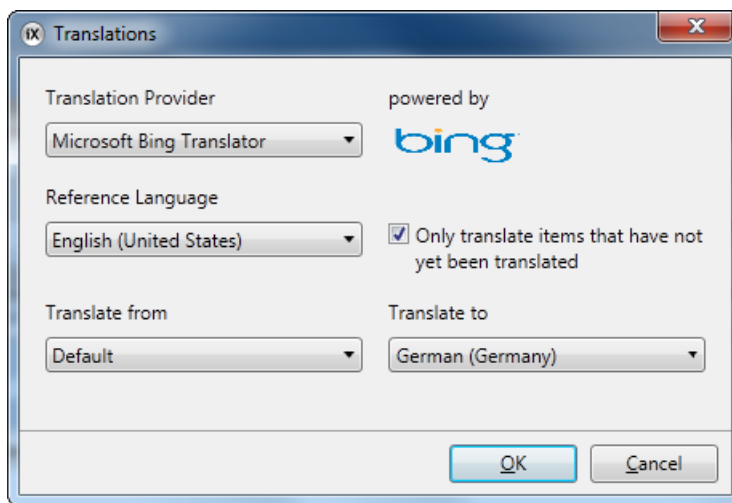
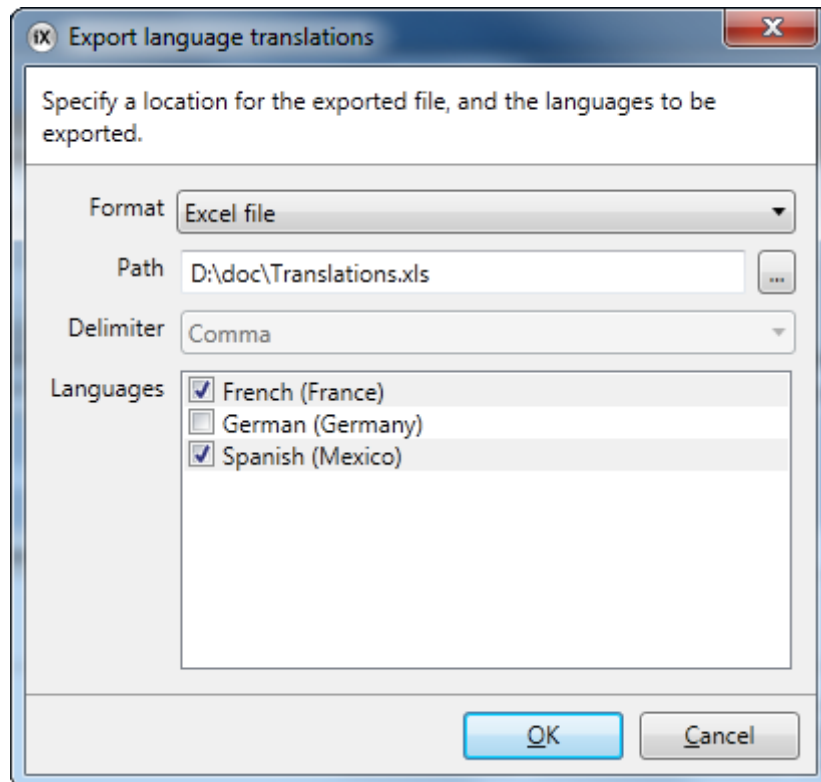| Parameter	Description
Translation Provider	Select translation provider among available providers. Microsoft Bing Translator is the only translation provider in iX Developer.
Reference Language	Select reference language; i.e. the language in which the project was created.
Only translate items that have not yet been translated	Unchecking the box replaces also already translated items in the project.
Translate from	Selection of source language for the translation operation; the default language (reference language) or one of the languages that have been added to the project.
Translate to	Selection of target language for the translation operation.

Note:
Since the translation tool accesses online translation providers, the translation operation may sometimes be paused or delayed due to an excess of online users.

17.5 Exporting Languages

Languages can be exported to a text file for e.g. translation.

1. Select **Export** from the **Import/Export** button.
2. Specify format, location, delimiter (for text file format) and language selection, and click **OK**.

Parameter	Description
Format	Select format of the exported file; text file or Excel file.
Path	Browse to the location for the exported file
Delimiter	If **Text file** format was selected, select delimiter from the drop-down list; comma, semicolon, space or tab
Languages	Select which languages to export

The exported file will also include the default language.

Note:
After having exported user/system texts with national characters to be edited in Excel, **65001: Unicode (UTF8)** format has to be selected in the Excel import dialog. Otherwise, national characters will turn out corrupt in Excel.

17.6 Importing Languages

After translation or other editing, the updated language file can be imported to the project again.

1. Select **Import** from the **Import/Export** button.
2. Specify location and language selection, and click **OK**.

Parameter	Description
Format	Select format of the file to import; text file or Excel file.
Path	Browse to the location of the file to import
Languages	Select which languages to import
Import Strategy	Selecting **Key** performs the import according to the **Designer** and **Object** columns in the multiple languages configuration page. **Reference Text** performs the import according to the **Text** column. If you have changed the texts of the default language after exporting the languages, it is recommended to select **Key**.

The imported file will also include the default language.

Note:
Make sure to make a backup copy of the project before importing languages, in case there are mismatches between the existing and the imported languages.

17.7 Hebrew Language

Hebrew is written from right to left, opposite to the way most modern languages are written.

iX Developer does not support Hebrew keyboards. It is suggested that you write your text entries using a Hebrew font in an external text editor that supports Hebrew keyboards, then copy the text entry into your iX Developer project.

The **Arial Unicode** is an example of a font that supports the Hebrew language.

18   Audit Trail

The Audit Trail function makes it possible to track operator actions.

The Audit Trail database can be exported in csv format using the **Database Export** action. It is recommended to use the database export method rather than the backup method.

Related information
Database Export

18.1 Logging Strategies

To meet FDA (Food and Drug Administration) demands, it is possible to log actions in a strict manner. FDA logging implies that no log entries will ever be overwritten. If the size of the audit trail database is close to being exceeded (approximately 80% of the limit), a notification window will be displayed, where the user is warned, and demanded to export the data. Alternatively, the size of the Audit Trail database can be increased. If the limit of the database is exceeded, the logging is terminated and the operator panel will be blocked for any further operator actions. This is indicated by an error message.

With circular buffering, the oldest log entry is overwritten when the database is full. No warning message is displayed when reaching the limit of the database.

The database can be exported using an action, regardless of logging strategy. The log entries are saved in the Audit Trail database, and the information can be displayed using the Audit Trail Viewer.

18.2 Using the Audit Trail Function

The function is enabled by selecting **Audit Trail** from the **Insert** ribbon tab. The configuration page opens in the desktop area. All actions available for logging are listed. It is possible to log all or only some actions. It is also possible to log tag changes only (without logging actions).

Parameter	Description
Log All / Log None	Click the buttons to quickly select all or no actions from the list
Logging strategy	Select FDA or Cyclic Buffering
Max size of database	The maximum size of the Audit Trail database

Value change on tags, performed by user actions, can be logged by selecting **Log to Audit Trail** in the tags configuration page. This option is enabled only if the audit trail function is activated in the project.

Related information
Others in *Adding Tags* section

18.2.1   Audit Trail Description

It is possible to add a description for each object that is logged to the Audit Trail database using the Property Grid.
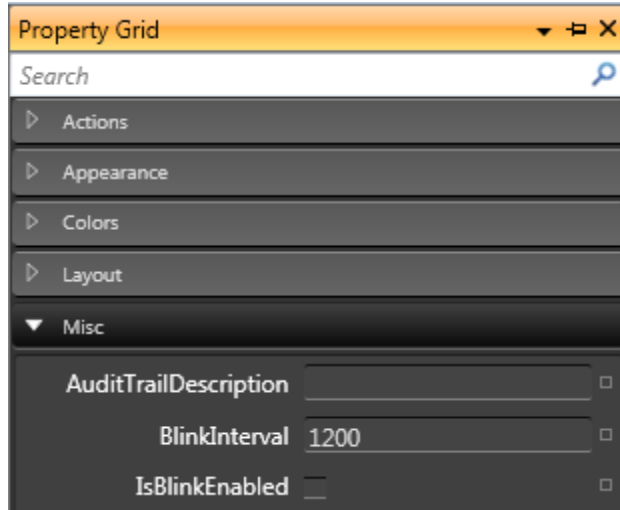

The description will be saved to the **Description** column of the Audit Trail database, and is limited to 255 characters.

18.3 Audit Trail Viewer

The Audit Trail Viewer is selected from the **Objects** group on the Home ribbon tab. It displays the information from the Audit Trail database.

Selecting **Configure Columns** on the **General** tab when the Audit Trail Viewer object is selected makes it possible to choose and configure columns to display in runtime.
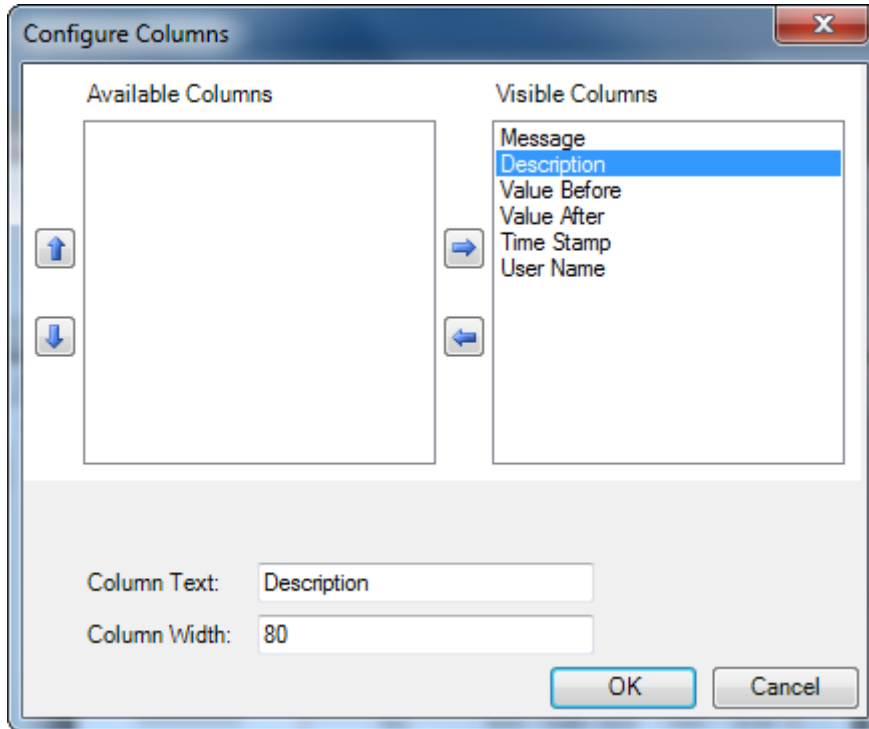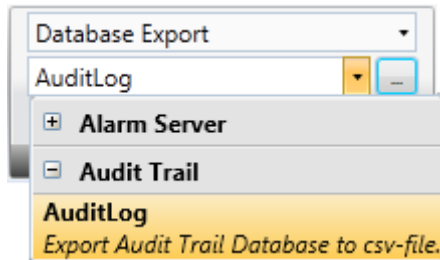
18.4 Audit Trail Export

An audit trail database can be exported as a .csv file and saved to a USB stick, an external memory card or to the project files folder. The export is configured through the **Database Export** action available from the **Database** action group.
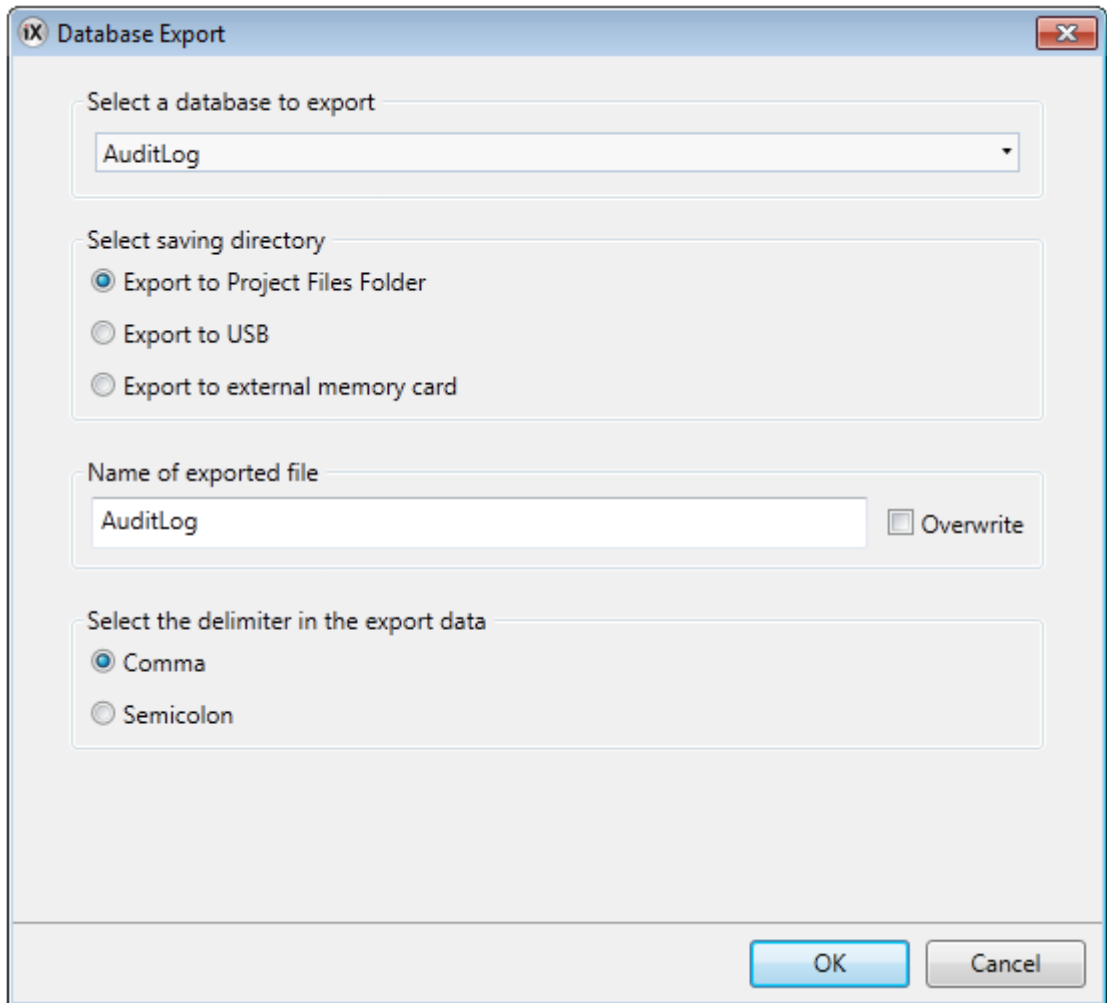

Note:
The export will not delete the audit trail log from the project.

| Related information |
|---|
| *Action Groups* |

18.4.1   Audit Trail Export from an Operator Panel Target

It is possible to select **export to USB**, **export to external memory card** or **export to the Project Files folder**, and also setting another name of the export file.

It is also possible to select the requested delimiter in the .csv file (comma or semicolon).

18.4.2   Audit Trail Export from a PC Target
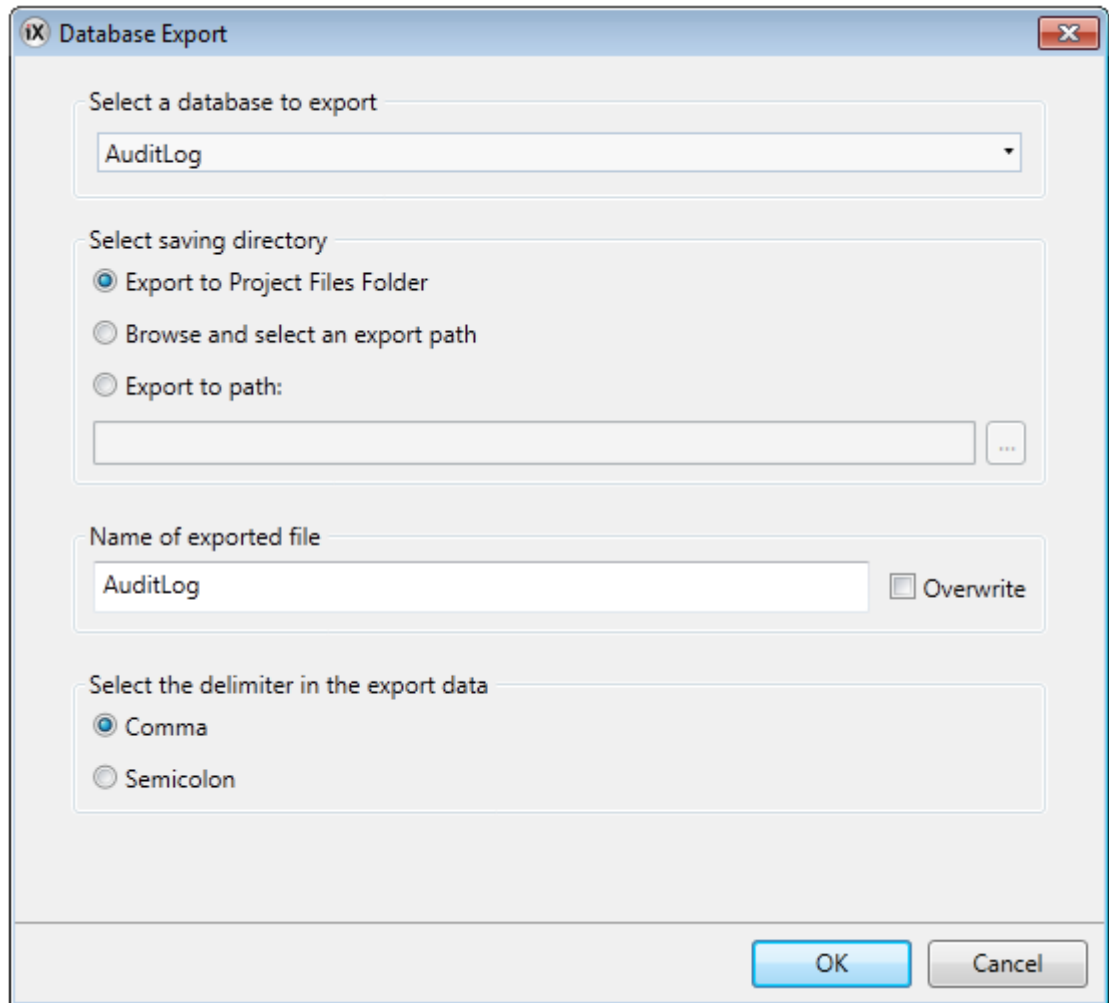
It is possible to select **Browse and Select an export path**, **Export to Path** or the **Export to Project Files folder**, and also setting another name of the export file.

The **Browse and Select an export path** option allows browsing for desired export destination in runtime. This may be useful if available folders in runtime are unknown when designing the project.

Select the **Export to Path** option when you want to set the path when designing the project. If the pre-defined path is not valid in runtime, the operator may browse the PC environment by clicking a browse button.

It is also possible to select the requested delimiter in the .csv file (comma or semicolon).

19   Text Library

With the text library function, text tables can be created, where values are linked to texts. The text library function can be used for presentation of each sequence step in sequence control. Another possible application is to present error codes, where an analog signal generates error codes connected to texts in a text library. The function can also be used to give analog values specific values, depending on the selected texts.

The text library function is activated by selecting it from the **Insert** ribbon tab.
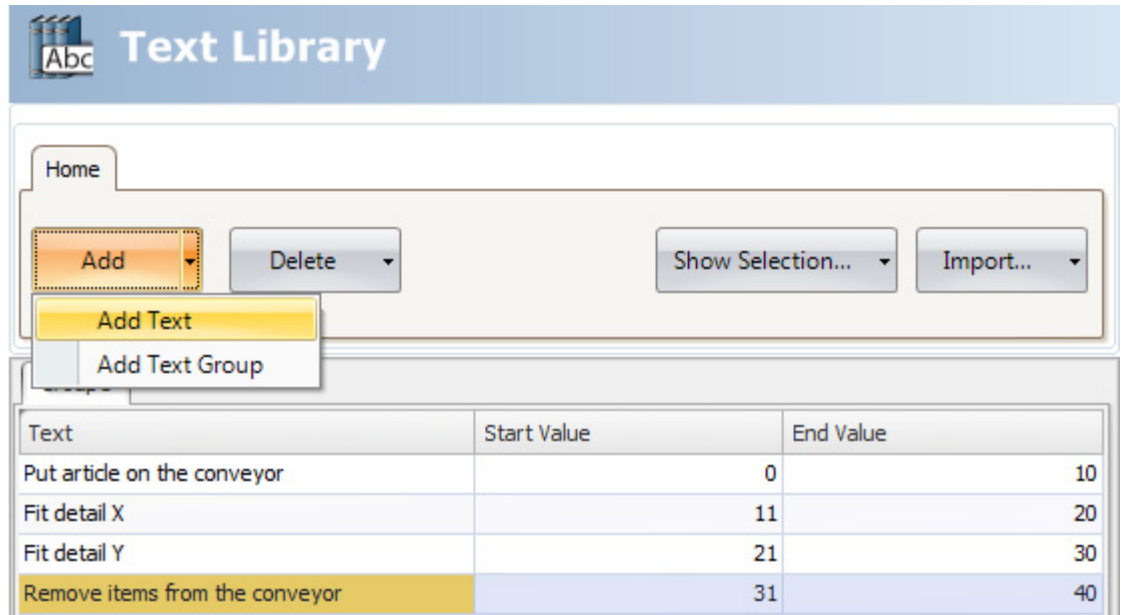

Texts and groups can be added from the **Add** button.

It is also possible to add start and end values for each message, that makes the text in the connected object change depending on the value of the tag.

19.1 Connecting Objects to Text Library Texts

Perform the following steps to connect an object to a text library text:

1. Select the object on the screen.
2. Click on **General** on the Dynamics tab.
3. Select the **Text** property, and select **Text Library Group Converter**.
4. Browse to the tag and select the text library group.
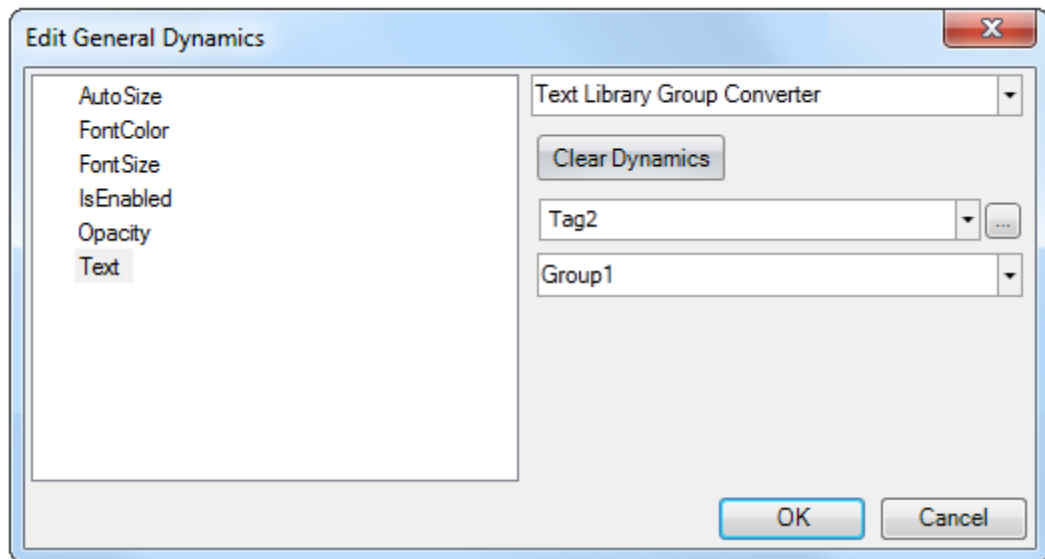
The object will now present the texts from the text library instead of the value.

19.2 Exporting and Importing Text Library Texts

The texts in the text library can be exported and imported using the **Import/Export** button in the Text library configuration page. The procedure is similar to exporting and importing tags.

| Related information |
|---|
| *Importing and Exporting Tags* |
| *Tag Import Example* |

20   Database Export

Some databases in the project can be exported as a .csv file to a USB stick, an external memory card or to the project files folder. The following databases are available for csv export: alarm server, audit trail, data logger and trend viewer.

Database export is not supported during project simulation.

20.1 Setting up Database Export

The export is triggered via an action, **Database Export**, in the **Database** group.
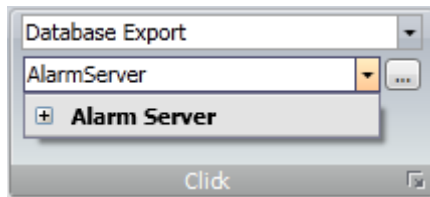
Individual database tables may be selected for export by clicking the + sign.

The default properties for the database export is exporting the database to the project files folder with the name as given in the project.

Note:
To facilitate exporting multiple databases at once, configuring multiple actions is recommended.

Clicking the **...** button next to the selected database opens a dialog from which the default database export properties may be altered.

20.1.1　Database Export from Operator Panel Target
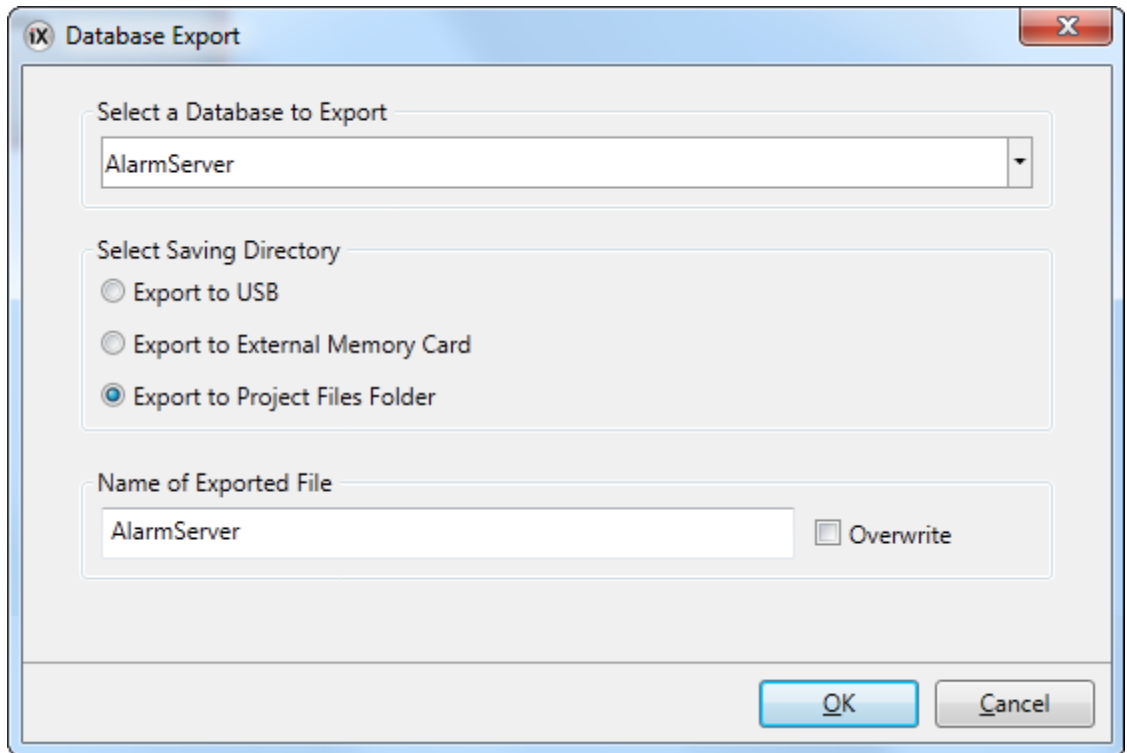
It is possible to select export to USB stick or external memory card, and also setting another name of the export file. Leaving the **Overwrite** box unchecked adds date and time to the export file name, thus avoiding overwriting an existing export file.

20.1.2   Database Export from PC Target
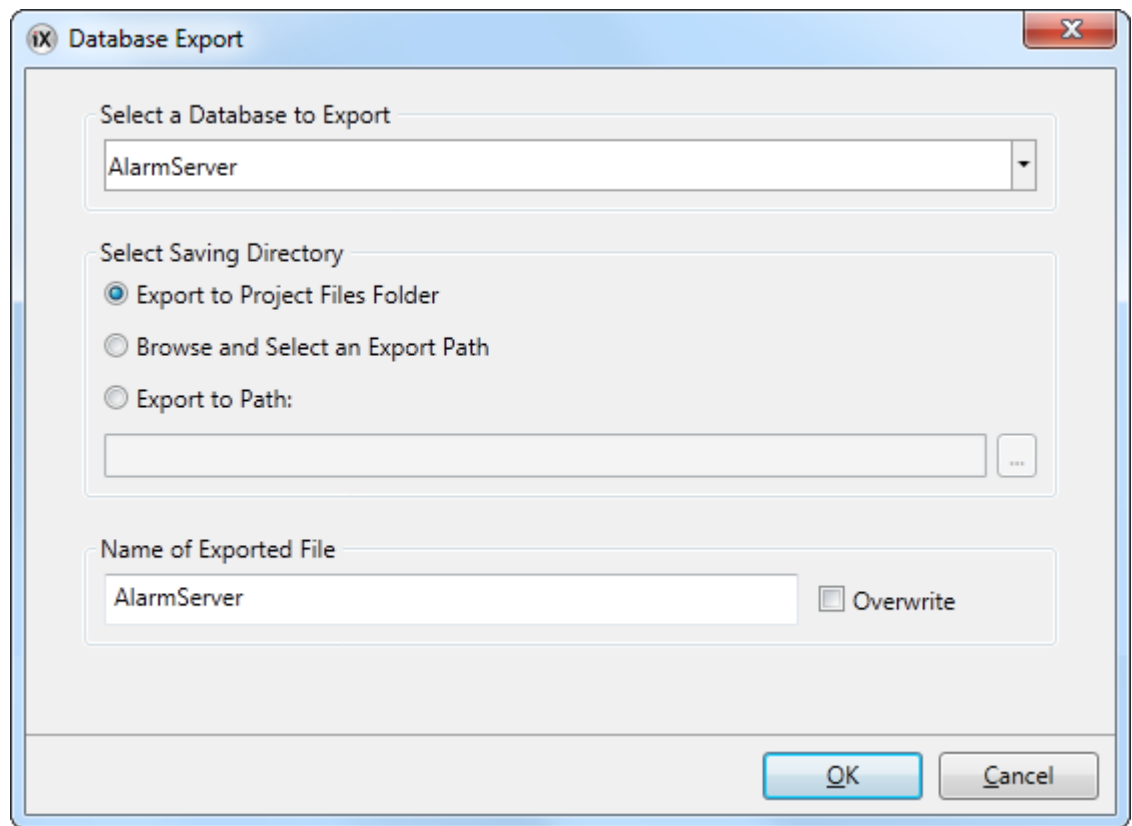
It is possible to set another name of the export file. Leaving the **Overwrite** box unchecked adds date and time to the export file name, thus avoiding overwriting an existing export file.

The **Browse and Select an Export Path** option allows browsing for desired export destination in runtime. This may be useful if available folders in runtime are unknown when designing the project.

Select the **Export to Path** option when you want to set the path when designing the project. If the pre-defined path is not valid in runtime, the operator may browse the PC environment by clicking a browse button.

Related information
Actions Ribbon Tab
Simulate
Multiple Actions

21   Operator Panel System Software

In the operator panel, there is a system program (operating system), which is stored in the operator panel's memory on delivery.

The system program can be replaced, that is, updated to a newer version, either via PC or via external memory card.

An Image Loader is used to download system programs to the operator panel via a PC. The Image Loader is available from the web site.

Note:
When updating an operator panel, it is important to ensure that power is not interrupted during the transfer.

21.1 Welcome Screen

When the operator panel is started, the welcome screen is displayed. The following items about the operator panel are listed:

- Size of internal memory card, if applicable
- IP address
- Image version

If there is a project in the panel, it will be loaded automatically.

If there is no project in the panel, touching the screen will display the Service menu.

Related information
Service Menu

21.2 Service Menu

The service menu for the operator panel can be accessed before a project is downloaded, or by setting the mode switches in mode 1000. The location of the mode switches is described in the installation manual for the operator panel.

21.2.1   Service Menu in an Empty Panel

When no project is loaded in the panel memory, the panel will boot, displaying the Welcome screen.

• Press anywhere on the panel display to enter the service menu.

21.2.2   Service Menu in TxA/TxB Panels

Perform the following steps to enter the service menu:

1. Apply power to the panel.
2. When the hourglass picture appears, press a finger on the screen and hold for approximately 20 seconds.
3. If the service menu is password protected, you will be prompted for a pin code. Enter pin code.
4. The touch calibration screen will display the following message:
   "Tap anywhere on screen or touch calibrate will start in 10 seconds."
5. Press finger once again to enter the service menu.

21.2.3   Service Menu in TA Panels with Project

When a project is loaded in the panel memory, the panel will automatically start to execute the project. To display the service menu at boot-up:

1. Disconnect the power supply from the panel.
2. Set the mode switches in mode 1000 (Service Menu mode).
3. Connect the power supply.

   When the panel boots up it will suggest to go into the **Touch Calibrate** mode. This ensures that it will be possible to recover from a bad calibration.
4. Press anywhere on the panel display to interrupt the timer and to enter the service menu.
5. Make desired settings in the service menu.
6. Disconnect the power supply from the panel.
7. Set the mode switches in mode 0000 (Run mode).
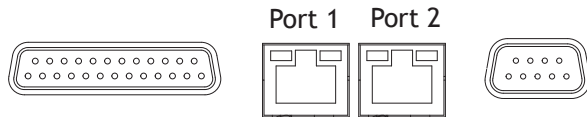8. Reconnect the power supply.

Related information
Service Menu Group

21.2.4   Service Menu Options

IP Settings

Select to obtain an IP address automatically via DHCP, or specify an IP address. The IP address can also be set during project transfer.

For panels with two Ethernet ports, select port according to the illustration below.

The port that is currently connected is recognized by a *.

Date/Time

Use the **Date/Time Settings** dialog to set the time zone, date and time for the panel.

Erase Project

The erase function will detect if the project is located in the panel memory or on a memory card. Pressing **Erase Project** will completely remove the project and all its components from the panel memory/the memory card.

Format Memory Card (for Some Models)

The format memory card function will detect if external and internal memory cards. Select which card to format and, in some cases, formatting alternatives.

Touch Calibrate

Select **Touch Calibrate** if the touch screen needs to be calibrated. Follow the instructions on the screen to perform a calibration.

Close

Press **Close** to exit the service menu.

21.3 Operator Panel Upgrade

The operator panel can be upgraded using the Image Loader application, and setting the mode switches in mode 0100. The location of the mode switches is described in the installation manual for the operator panel .

21.3.1 Requirements

Updating of the operator panel system program requires the following:

- A PC with Image Loader application.
- An Ethernet connection between the PC and the operator panel.
- The operator panel IP address.
- The Image file from the web site.

21.3.2 Transfer Image

1. Double-click on the executable Image Loader file to start the transfer program.
2. Follow the instructions.

Transfer Procedure

The image transfer procedure is completely menu-driven. The operator panel will be ready for transfer directly afterwards, provided that all steps are performed and completed.

The following steps outline the transfer procedure:

1. Disconnect the power supply from the panel.
2. Set the mode switches in mode 0100 (Image Load mode).
3. Reconnect the power supply.
4. Select the operator panel to upgrade by entering its IP address.
5. Click **Upgrade**.
6. When the upgrade is finished, disconnect power from the operator panel.
7. Set the mode switches in mode 0000 (Run mode).
8. Reconnect power to the operator panel.

Note:
When updating an operator panel, it is important to ensure that power is not interrupted during the transfer.

22   Multiple Controllers

It is possible to use multiple drivers with one operator panel. This means that the operator panel can communicate with different controllers simultaneously. The controllers can be connected to the serial ports on the operator panel and via the Ethernet port.

Note:
If the same tag is connected to addresses in more than one active controller, there is no way to control from which controller the value is read in runtime.

22.1 Adding a Controller

Click on **Tags** in the Project Explorer, select the **Controller** tab and click **Add** to add a new controller in the current project, and select which controller model and driver that is to be used.

Tags are added and controller is selected on the Tags tab.

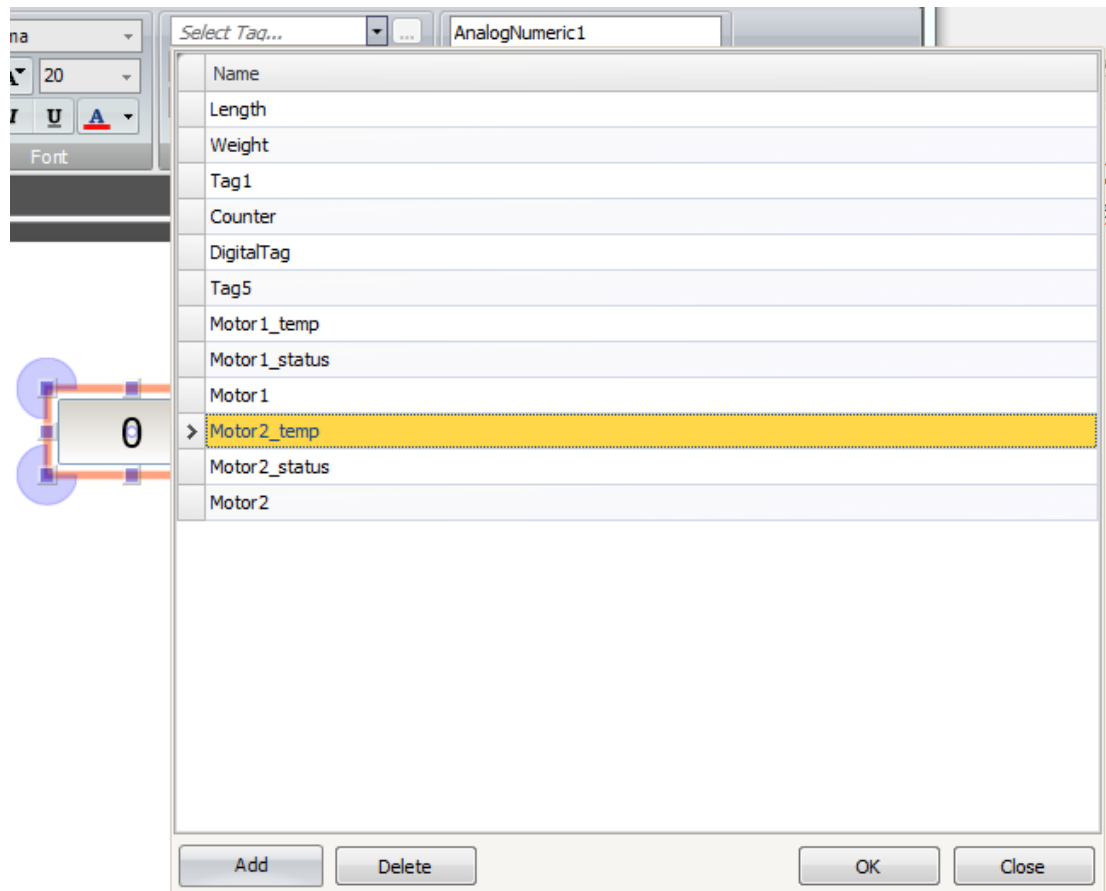| Tag			Controllers		
Name	Data Type	Access Right	Data Type	DemoController	SiemensController ▲
Length	DEFAULT	ReadWrite	INT16	D2	
Weight	DEFAULT	ReadWrite	INT16	D0	
Tag1	DEFAULT	ReadWrite	INT16	D11	
Counter	DEFAULT	ReadWrite	INT16	C0	
DigitalTag	DEFAULT	ReadWrite	BIT	M4	
Tag5	DEFAULT	ReadWrite	INT16	D15	
Motor1_temp	DEFAULT	ReadWrite	INT16	D20	
Motor1_status	DEFAULT	ReadWrite	BIT	M2	
Motor1	DEFAULT	ReadWrite	BIT	M3	
Motor2_temp	DEFAULT	ReadWrite	INT16		V1
Motor2_status	DEFAULT	ReadWrite	BIT		Y2
Motor2 ⋯	DEFAULT	ReadWrite	BIT		Y3

| **Related information** |
|---|
| *Adding a Controller* |

22.1.1 Addressing

Addressing of the tags in the controllers is carried out according to the help file for each driver.

When assigning a tag to an object, all tags, regardless of controller connection, are available for selection.
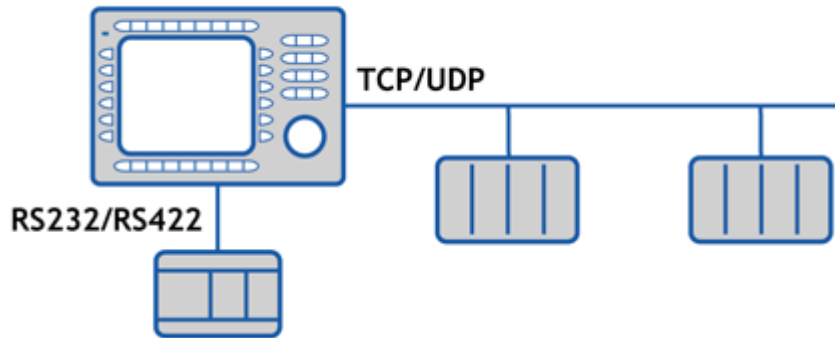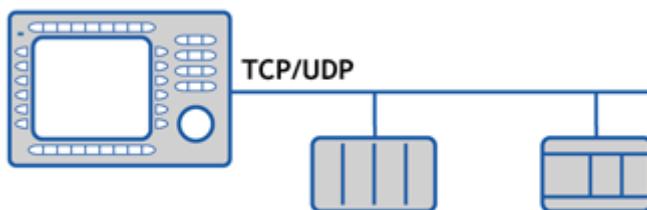
22.1.2 Examples

The following examples show configurations where multiple drivers are used in the operator panel.
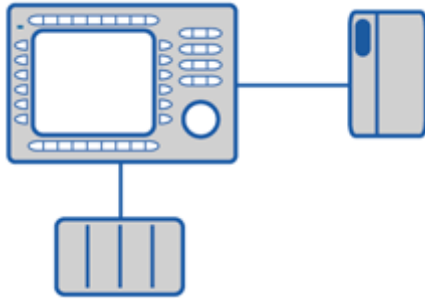
Operator panel communicating with one controller and one frequency converter.

Operator panel communicating in series with one controller via the Ethernet and with controllers of another make.

Operator panel communicating with two controllers of different makes via the Ethernet.

23   Trouble Shooting

This section describes tools to use in case of unexpected behavior in iX Developer.

23.1 Project Build Failure

If project build fails, check that the absolute path to the project, including its components, does not exceed 260 characters.

23.2 Backup Project Issues

Make sure that project that is fetched using the Backup command is identical to the current iX Developer project; i.e. has the same name and is of the same size.

Related information
Upload Database

23.3 Invalid Names

If using system device names when naming objects such as controllers, screens, recipe managers, data loggers and script modules, an error message will be displayed, stating that this is an invalid name. Examples of system device names: AUX, COM1, CON, LPT3, NUL, PRN.

Related information
Name Group

23.4 Adobe Reader not Recognized

In order for the PDF Viewer object to recognize that Adobe Reader is installed in the development PC as well as in the target, the setting **Display PDF in browser** has to be enabled in Adobe Reader.

1. Open Adobe Reader and select **Edit/Preferences/Internet**.
2. Check the **Display PDF in browser** box.

23.5 Expression Cannot Be Found

If an error indicating that an expression cannot be found is generated at project validation, make sure that objects inserted from the Component library uses global expressions only.

Related information
Using Expressions

23.6 Performance Issue Related to Graphic Card

When designing iX Developer projects, the CPU load may suddenly increase when performing actions such as moving objects on the screen or hovering over buttons, depending on which graphic card is used.

This problem may be solved by adjusting the hardware acceleration of the graphic card.

In Windows XP, follow the steps below to make needed adjustments:

1. Double-click on the **Display** icon in the Windows control panel.

2. Select the **Settings** tab and click on the **Advanced** button.

3. Select the **Troubleshoot** tab and decrease or turn off the hardware acceleration by dragging the slider.

Related information
System Requirements and Limitations

23.7 Error Related to Third Party Controls

When opening a project including third party controls, the third party controls have to be included in the iX Developer third party control folder. Otherwise the project may terminate, or unexpected error messages may be displayed.

23.8 Performance Issue When Navigating in Script Tree View

If you are working with a large iX Developer project, especially a project with many tags and alarms, you may find that navigating the script tree view is slow.

The performance can be improved by saving the project.

Related information
Script

Beijer

ELECTRONICS