Release Notes V5.4

2016 03 16

Table of Contents

Upgrading to Proview V5.4.0

This document describes new functions i Proview V5.4.0, and how to upgrade a project from V5.3.0 to V5.4.0.

New functions

Ge Custom color palette

The new custom color palette is positioned below the ordinary color palette. It contains 90 colors that can be defined by the user. By double clicking on a color, the color selector is opened where the color is specified. The colors in the custom color palette can be used as fill color, border color, text color and in color dynamics as any other color. If a custom color is modified, all objects drawn with this color is updated with the new color.

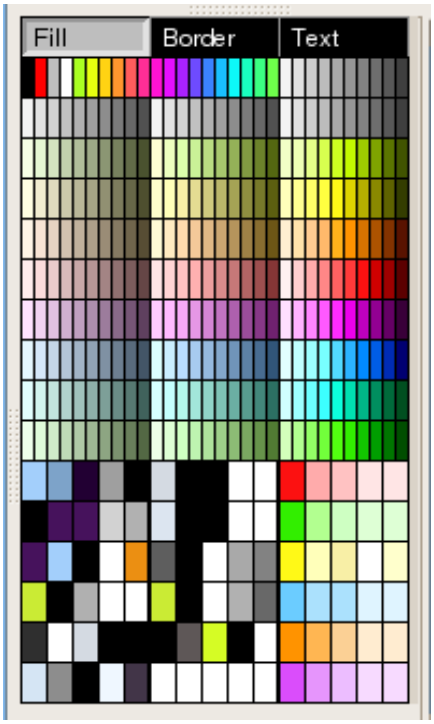


Fig Custom color palette

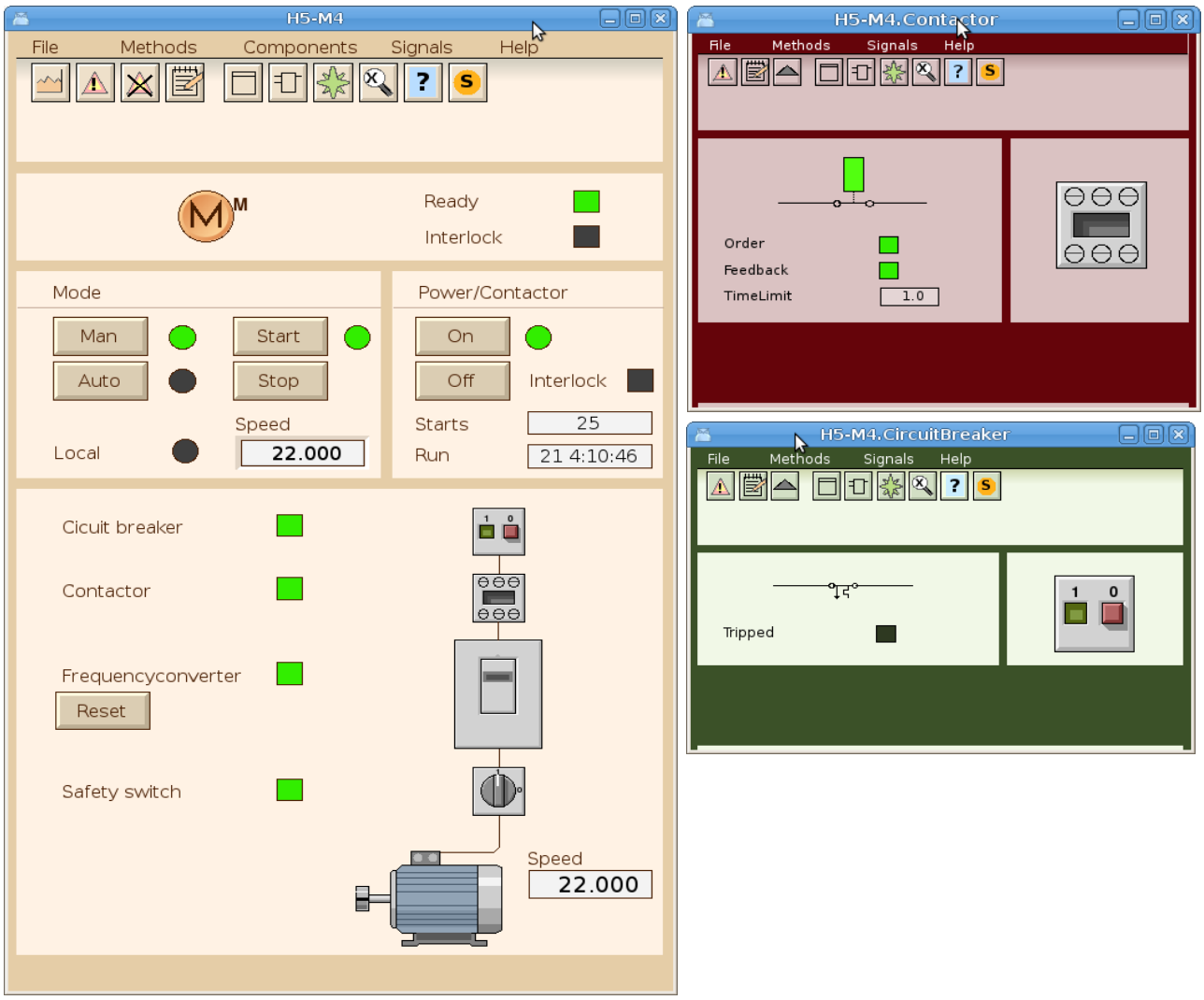
it is possible to save the colors in custom color palette to file and load them into other graphs. This is done by activating File/CustomColors/Save and File/CustomColors/Load in the menu. The colors are stored in pwrc-files on $pwrp_exe.

It's also possible to use a common color file for several graphs in a project. If a color-file is specified in ColorTheme in Graph attributes, this file will be loaded automatically when the graph is opened. When a color is adjusted and stored, it will affect all object drawn with this color in all graphs with this color-file. In this way the colors in the graphs can easily be adjusted.

Fig Color selector

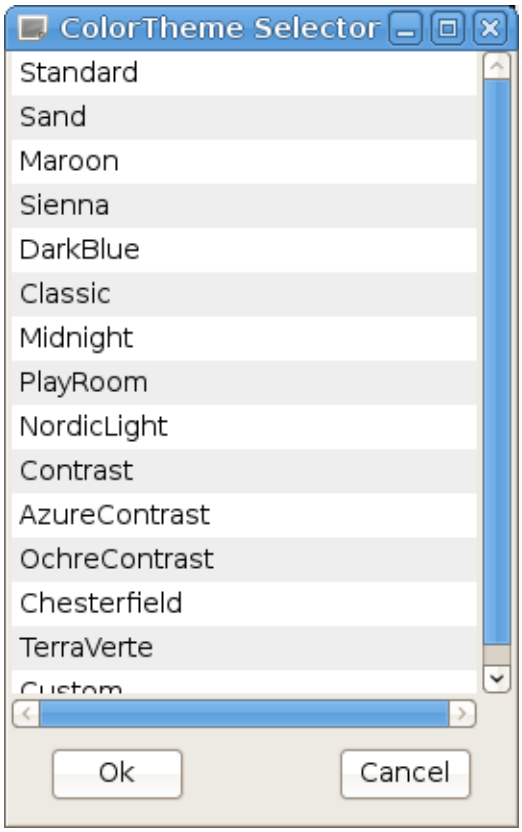
Object graphs updated

All object graphs are redrawn and updated. The graphs are drawn with color themes and follow the currently selected theme.



Color themes

The color theme is selected by the operator from Functions/View/Color Theme in the operator window menu.



The color theme will specify the colors for the operator window, object graphs, and possibly also project graphs if these are drawn with a colortheme table. Here are some examples of color themes.

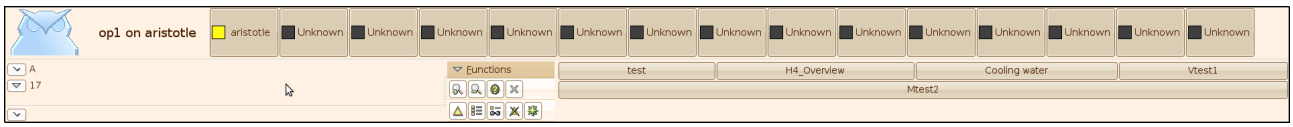
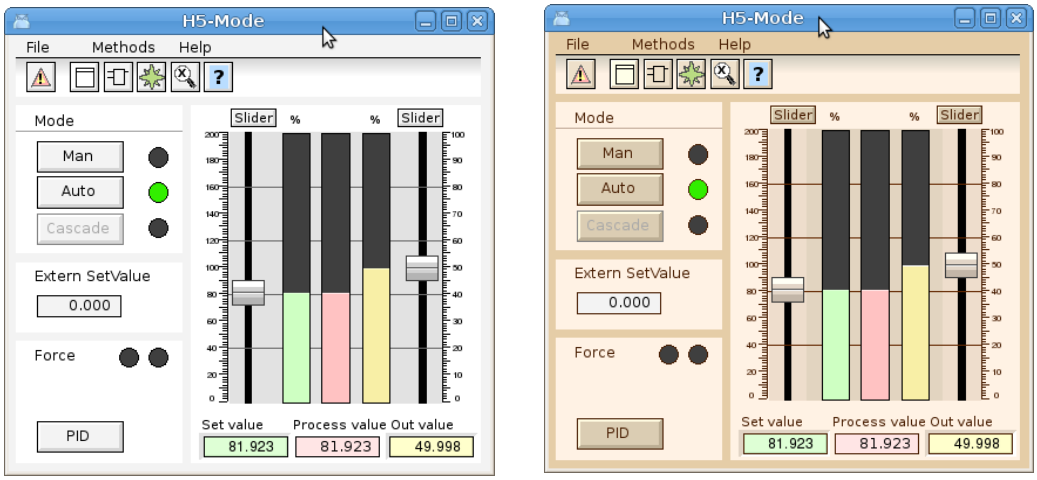
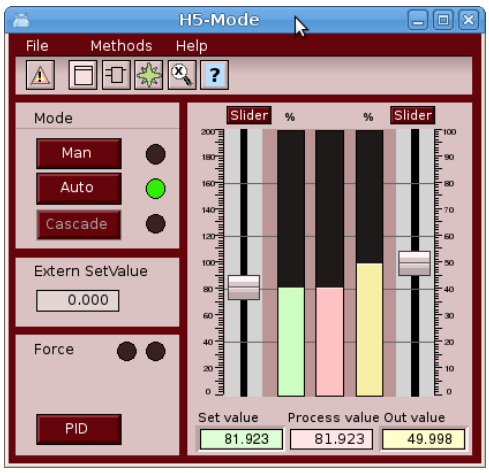


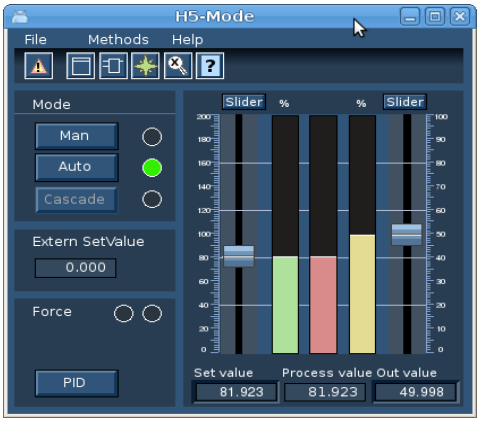
Fig Operator window with Sand color theme

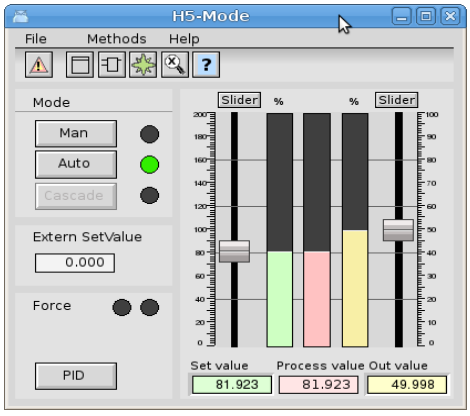


Standard and Sand

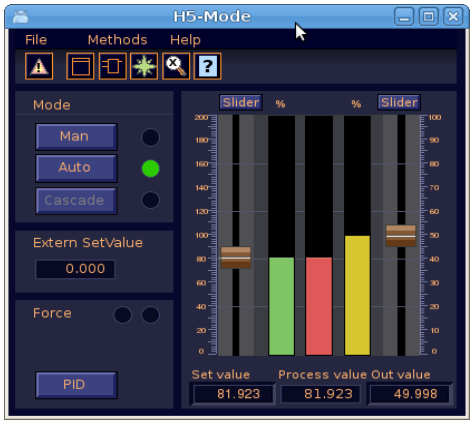


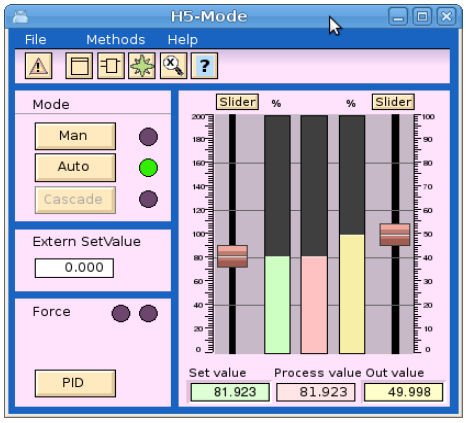
Marron and Sienna



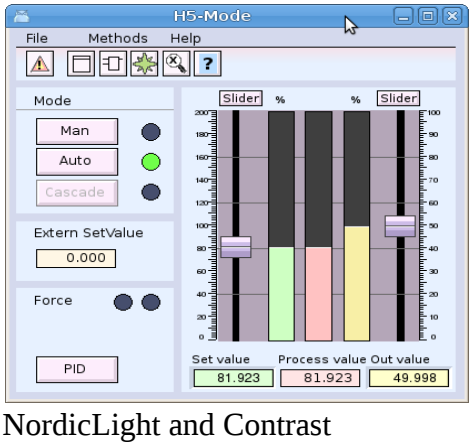


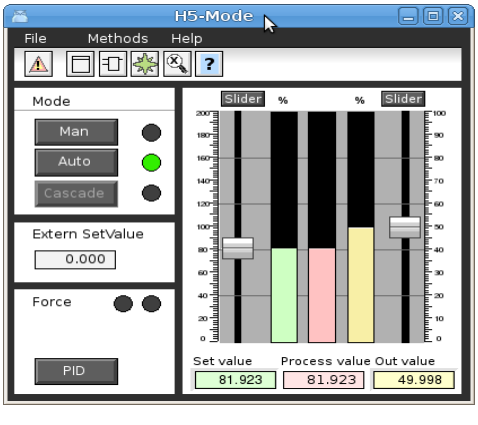
DarkBlue and Classic



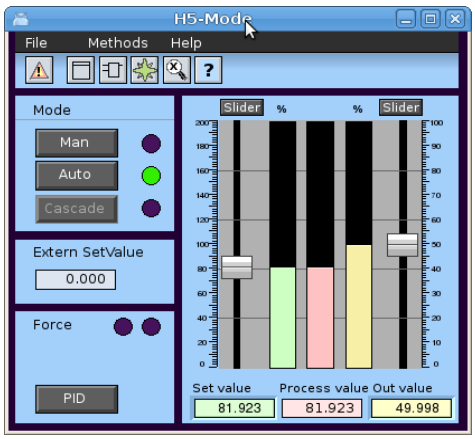


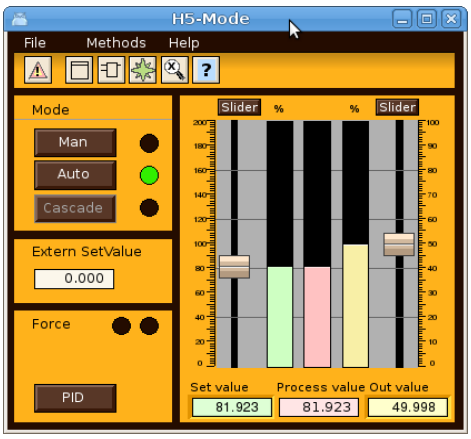
Midnight and Playroom



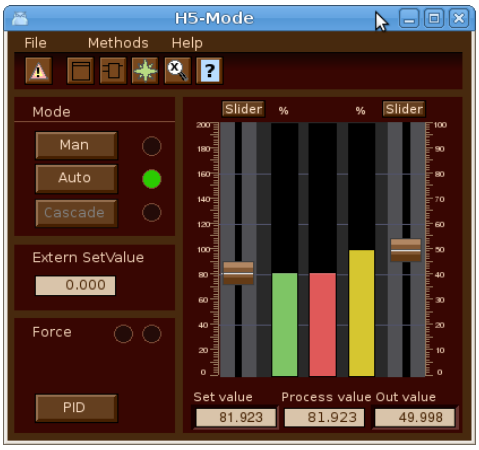


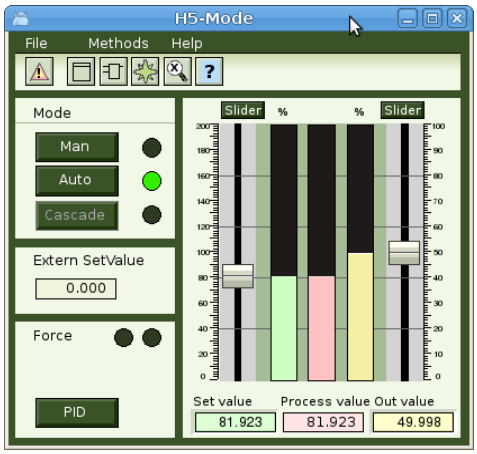
NordicLight and Contrast



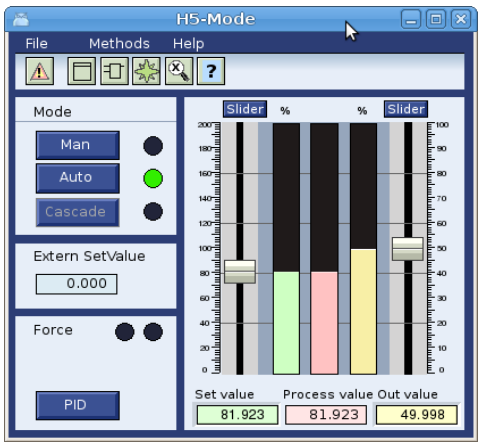


AzurContrast and OchreContrast





Chesterfield and TerraVerte



Polar

Graphs drawn with color theme uses color tables for custom colors where

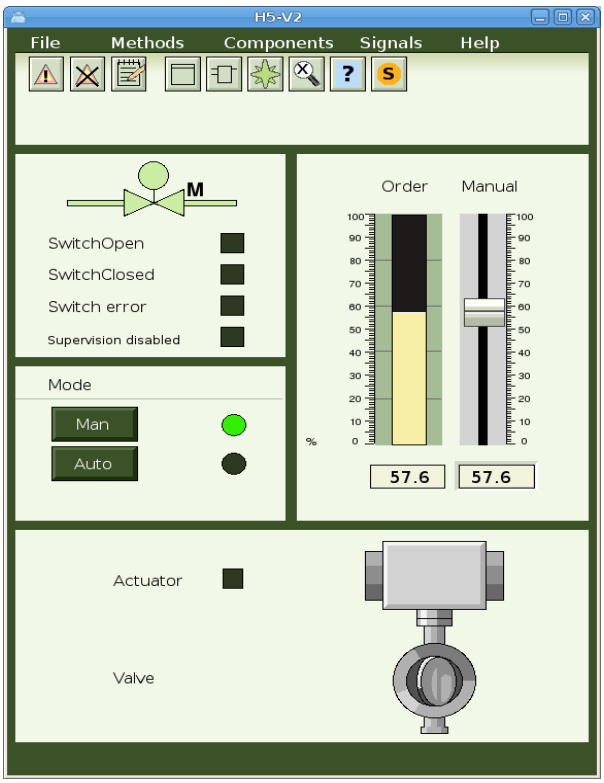
Using color theme in projects graphs

The color themes can also be used when drawing graphs in a project. These graphs will follow the appearance of the color theme selected by the operator, but it requires that specific colors in in the custom color palette is used when drawing different objects.

A color theme is loaded from File/ColorTheme/Select in the Ge menu. The colors for the theme is then loaded into the custom colors palette. Every color in the custom color palette is adjusted for a specific purpose. The first color for example is the background color for the graph, and the fifth color should be set on texts on the background. By placing the cursor on a color in the palette, the purpose of this color will be written in a field below the palette.

The color theme has support for specific elements in a graph. In the example below the background color is lightgreen. The dark green lines are called delimiter areas. In this case it is rectangles the divides the graph into different sections. It's also possible to use if for larger areas with text and indicators. At the top there is a menu bar with pulldown menues, and below this a gradient to make a shadow below the menu. Object graphs should contain a method toolbar positions below the menu. Other elements are bars, axises, buttons, indicators, texts on the background, and texts on a delimiter area, sliders, value fields and input fields, tables, diagrams etc.

For more complex elements that are are not suited for a change of fill color, as the gray actuator and valve below, the dynamic ColorThemeLightness can be used to adapt the lightness of these elements to the lightness of the theme.



Description of color theme colors

Num	Name	Description
1	Background	Should be set as background color for the graph. Light green in the example above.
2	Background gradient	Can be used for gradients to the background color, eg the gradient under the menu above.
3	Delimiter area	Color of delimiter areas. The dark green areas.
4	Delimiter lines	Color of lines that divides the background, eg the line under Mode above.
5	Text/Lines on background	Color of texts and lines on the background.

16	Indicator border color	Should be set as border color on indicators.
17	Indicator low color	Low color for indicator
18	Indicator on delimiter low color	Low color for an indicator on a delimiter area.
19	Slider color	Color for sliders.
20	Slider background color	Color for slider background.

31	Limit switch high color	Color for limit switch when it's high.
32	Limit switch low color	Color for limit switch when it's low.
33	Limit switch border color	Color for limit switch border.
34	Text/Lines on delimiter	Color for text or lines on a delimiter area.
35	Button active color	Color to indicate that a button is active.

46	Symbol fill color	Color that can be used component symbols.
47	Symbol border color	Color for symbol border.
48	Symbol low color	Color when symbols is low.
49	Symbol empty color	Color that can be used to indicate that a valve in closed or empty.
50	-	

61	Menu fill color	Fill color for menu bars and menus.
62	Menu text color	Color for menu texts.
63	Toolbar fill color	Color for method toolbar in object graphs
64	Toolbar border color	Border color for method toolbar.
65	Toolbar text color	Text color for method toolbar.

Ge dynamics ColorThemeLightness

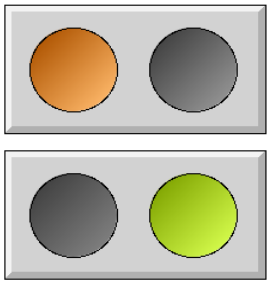
A color theme has a lightness property that is negative for dark themes and positive for light themes. The theme Midnight has for example lightness -5 while NordicLight has 1. Adding dynamic ColorThemeLightness to an object will increase or decrease the lightness of the object dependent on the colortheme lightness value. The ColorThemeLightness dynamic has no properties.

Ge dynamics DigBackgroundColor

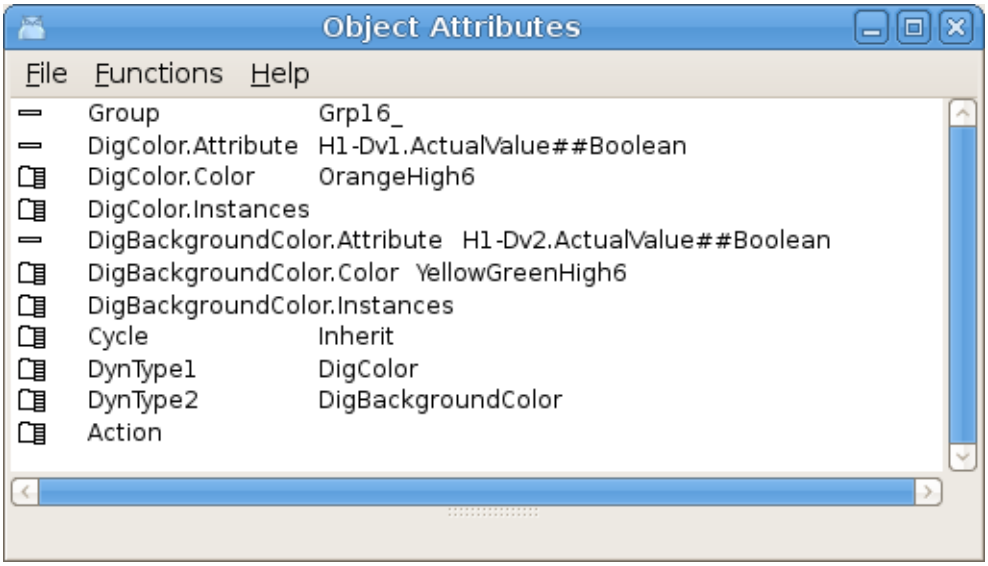
The DigBackgroundColor dynamics makes it possible to set dynamics on the new background color. DigBackgroundColor works analogous with DigColor, but will changes the background color instead of the fill color. The DigBackgroundColor has several instances and thus can be connected to several digital signals, and change between several colors.

DigBackground color makes it possible to modify the color of two parts of a subgraph or group

independently of each other. In the example below, the gray rectangle is drawn with fixed gray color, the left circle has the fill color, and the right background color. Drawing a circle with background color is achieved by setting the fill_eq_background property of the circle.



When setting dynamic DigColor and DigBackgroundColor to the group, the DigColor dynamic will affect the left circle drawn with fill color, and the DigBackgroundColor dynamics will affect the right circle drawn with background color.



Ge dynamics DigSwap

DigSwap will disconnect all subscriptions in the graph and reconnect them again. This is useful when subscriptions by reference is used with the &() syntax, eg &(H1-Plate.Ref).Length##Float32. In this example H1-Plate.Ref is an attribute reference that points to an object containing the Length attribute. If the reference in H1-Plate.Ref is changed, the subscriptions has to be reconnected to the new reference, and this is done by DigSwap. DigSwap is connected to a digital signal that will activate the swap.

Store and recall object properties

A new function in the Ge editor to store the properties for the currently selected object, and later recall these properties to another object, is added. The function is activate from 'Edit/Objects Attributes Store (Shift+Control A)' and 'Edit/Object Attributes Recall (Shift+Control D)' in the menu. The function is the same as Store and Recall in the object editor. The advantage is that the object editor doesn't have to be opened. Properties and dynamics can be copied by first selecting the source object and pressing Shift+Control A, and then selecting the target object and pressing Shift+Control D.

New properties in Ge objects

Rectangle properties

Rectangle has the new properties background color, fill_eq_background and bgcolor_gradient.

Background color

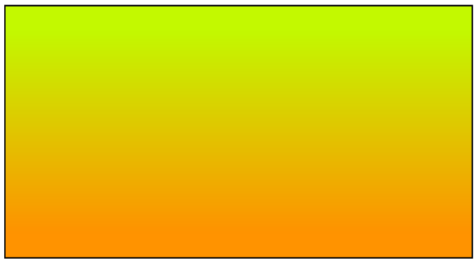
The background color can be set to a rectangle in the Ge editor by selecting the rectangle and clicking with Shift/Ctrl on a color in the palette.

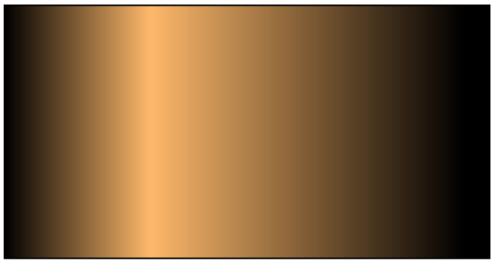
fill_eq_background

When fill_eq_background is set, the rectangle will be filled with the background color instead of the fill color. If the rectangle belongs to a subgraph or group, it will be filled with the background color for the sugraph or group.

bgcolor_gradient

When bg_color_gradient is set, any gradient will drawn between the background color and the fill color. In the left example below the background color for the rectangle is yellowgreen and the fill color is orange. In the right example background color is black, and fill color orange.





Arc properties

Background color

The background color can be set to an ac in the Ge editor by selecting the rectangle and clicking with Shift/Ctrl on a color in the palette.

fill_eq_background

When fill_eq_background is set, the arc will be filled with the background color instead of the fill color. If the arc belongs to a subgraph or group, it will be filled with the background color for the sugraph or group.

Polyline properties

Background color

The background color can be set to a polyline in the Ge editor by selecting the polyline and clicking with Shift/Ctrl on a color in the palette.

fill_eq_background

When fill_eq_background is set, the polyline will be filled with the background color instead of the fill color. If the polyline belongs to a subgraph or group, it will be filled with the background color for the sugraph or group.

fill_eq_bglight

When fill_eq_bglight is set, the polyline will be filled with a lighter tone of the background color.

fill_eq_bgshadow

When fill_eq_bglight is set, the polyline will be filled with a darker tone of the background color.

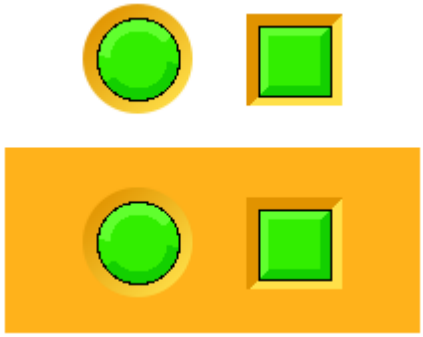
Subgraph and Group properties

Background color

All rectangles, arcs and polyline that is part of a subgraphs or group, that has the properties fill_eq_background, fill_eq_bglight or fill_eq_bgshadow set, will be drawn with the background color for the subgraph or group.

The background color for a subgraph or group is set by selecting the subgraph or group, and clicking with Shift/Ctrl on a color in the color palette.

In the left example below, the green round indicator has a surrounding yellow arc with fill_eq_background set and with diagonal gradient. The square indicator has two yellow poly lines around it, one with fill_eq_bgshadow, and one with fill_eq_bglight set. By setting the background color of the indicators, the background components can be adapted to the current background of the graph.



Ge object name

The name of a Ge object is displayed in the object editor. It can also be changed there. The possibility to change an object name from the menu 'Edit/Change name' is removed.

Ge search object

The menu entry 'Edit/Search object' in the Ge editor, makes it possible to search an object by name. As some error messages contains the name, the search function makes it easier to locate the error.

New Subgraphs

A number of new subgraph with colors adapted to the color theme palette is added. They can be found under the ColorTheme map.

New Ge commands

customcolor write

Write custom colors to file.

> customcolor write /file=

customcolor read

Read custom colors from file.

> customcolor read /file=

customcolor print

Print rgb-values for custom colors.

> customcolor print

customcolor set

Set properties lightness and isdefault to current color theme.

> customcolor set /lightness= /isdefault

New Ge script functions

GetFirstObject()

Get the first object in a graph.

GetNextObject()

Get the next object in a graph.

GetObjectAttribute()

Get an attribute value of an object.

GetObjectBorder()

Get object border property.

GetObjectBorderColor()

Get the border color of an object.

GetObjectClass()

Get the subgraph of an object.

GetObjectDynType()

Get dynamic and action type of an object.

GetObjectFill()

Get object fill property.

GetObjectFillColor()

Get the fill color of on object.

GetObjectGradient()

Get the gradient type of an object.

GetObjectName()

Get the name of an object.

GetObjectShadow()

Get the shadow property of an object.

GetObjectText()

Get the text of a text object.

GetObjectTextColor()

Get the text color of an object.

GetObjectType()

Get the object type.

GetRgbColor()

Get rgb values for a color.

GroupGetFirstObject()

Get first object in a group.

GroupGetNextObject()

Get next object in a group.

Reload()

Read the current graph from file.

SetObjectAttribute()

Set an attribute value of an object.

SetObjectBackgroundColor()

Set the background color of an object.

SetObjectFill()

Set the fill property of an object.

SetObjectShadow()

Set the shadow property of an object.

SetObjectTextColor()

Set the text color of an object.

SetRgbColor()

Set rgb values for a color.

New Xtt script functions

GetUser()

GetUser() returns a string with the current Proview user.

GetPrivileges()

GetPrivileges() returns the privilege mask for the current user.

Web configuration

The new class OpPlaceWeb is used to configure web pages. Attributes to configure the buttons for language, login, alarm and event list, event log, runtime navigator, project and Proview help is moved from the EventHandler object to the OpPlaceWeb object. Several web pages with different menues can be configured by creating several OpPlaceWeb objects with different filenames.

Buttons for Ge graphs and links are as before configured with WebGraph and WebLink objects, but they should now be positioned under the OpPlaceWeb object.

The WebHandler object is still used to configure the web server processes, rt_webmon, rt_webmonmh and rt_webmonelog.

Android app configuration

The OpPlaceApp object configures the android app. AppGraph and AppLink objects, previously configured under the WebHandler object, should now be configured under the OpPlaceApp object.

PID and CompPID modification for long integration times

The PID and ComPID objects are modified to avoid problems with remaining error when long integration times are used together with short scan times. The OpWindup and AbsOut attributes are changed from Float32 to Float64.

TimeMean, cumulative mean value

TimeMean is a plc function object that calculates the cumulative mean value over time.

QCom monitor update

The QCom monitor that handles all communication between nodes has been updated and adapted to higher network speeds.

Export buffer limit

During times of heavy load or bad network when outgoing messages can't be emitted in the pace they are entering the qcom monitor, they are buffered in an export queue. Previously there was no limit on this queue which could cause the network to go down on all links. Now this queue is limited and the queue will be purged when the limit is reached. The limit is configurable and can be set in NodeConfig.AComExportBufQuota. The limit, current usage and number of executed purges can be seen in the qcom node picture in rt_rtt.

Acknowledge delay

All QCom messages are sent with acknowledge to verify that the message has reached the target. If no ack is received within a certain time, the message is resent. Previously there was a delay of 1 tick (10 ms) before the ack was sent. This delay time is now configurable in NodeConfig.QComAckDelay.

Segment size

Large QCom messages are divided into segments. Previously the segment size was hard coded to 8192 bytes. It can now be configured in NodeConfig.QComSegmentSize. Note that all communicating nodes must have the same segment size.

Min and max resend time

When a messages is sent and no ack is received within the min resend time, the message will be resent. This time the timeout time is doubled, and if still no ack is received, the timeout time is doubled again. This continues until the timeout time reaches the max resend time, where the link is taken down. The min and max resend time previously had default values of 0.5 and 10 s which could be increased with configuring attributes in the NodeConfig object. Now they can also be decreased. The default value for min resend time is changed to 0.05 s.

rt_rtt show qnode picture updated

The rt_rtt qnode picture, opened from System/Communication/QCom/QCom Nodes, is updated and now displays the Round trip time in seconds. The segment size, acknowledge delay and export buffer quota and usages are also displayed. An error counter for segment sequence errors is also added.

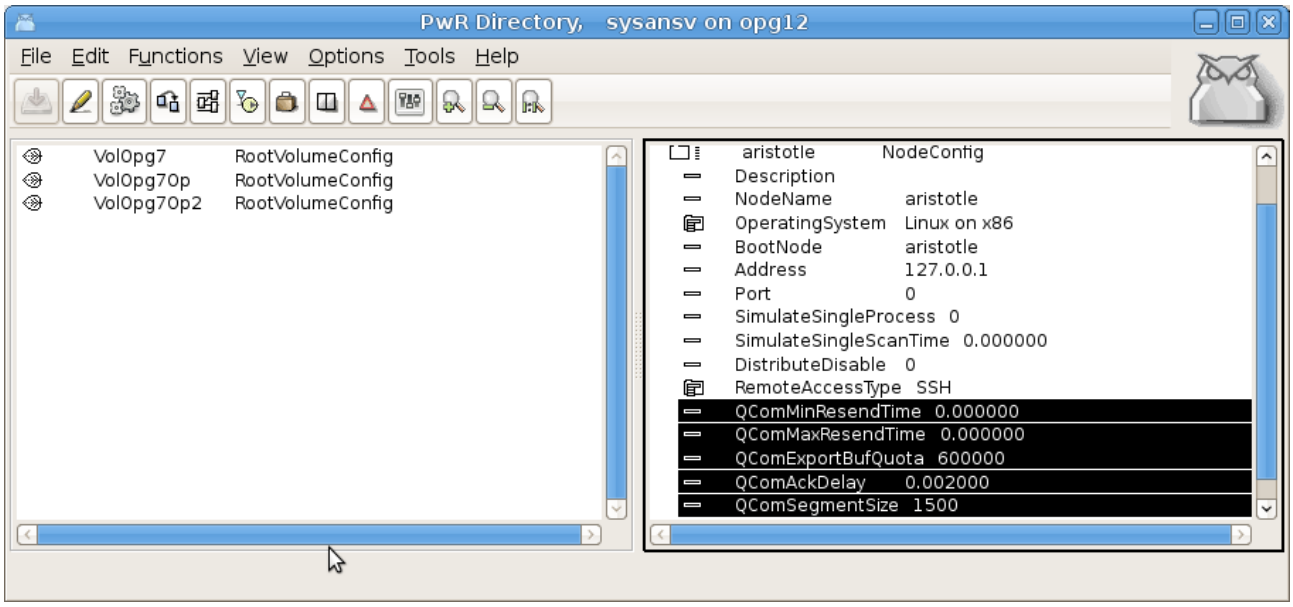


Fig QCom settings in the NodeConfig object

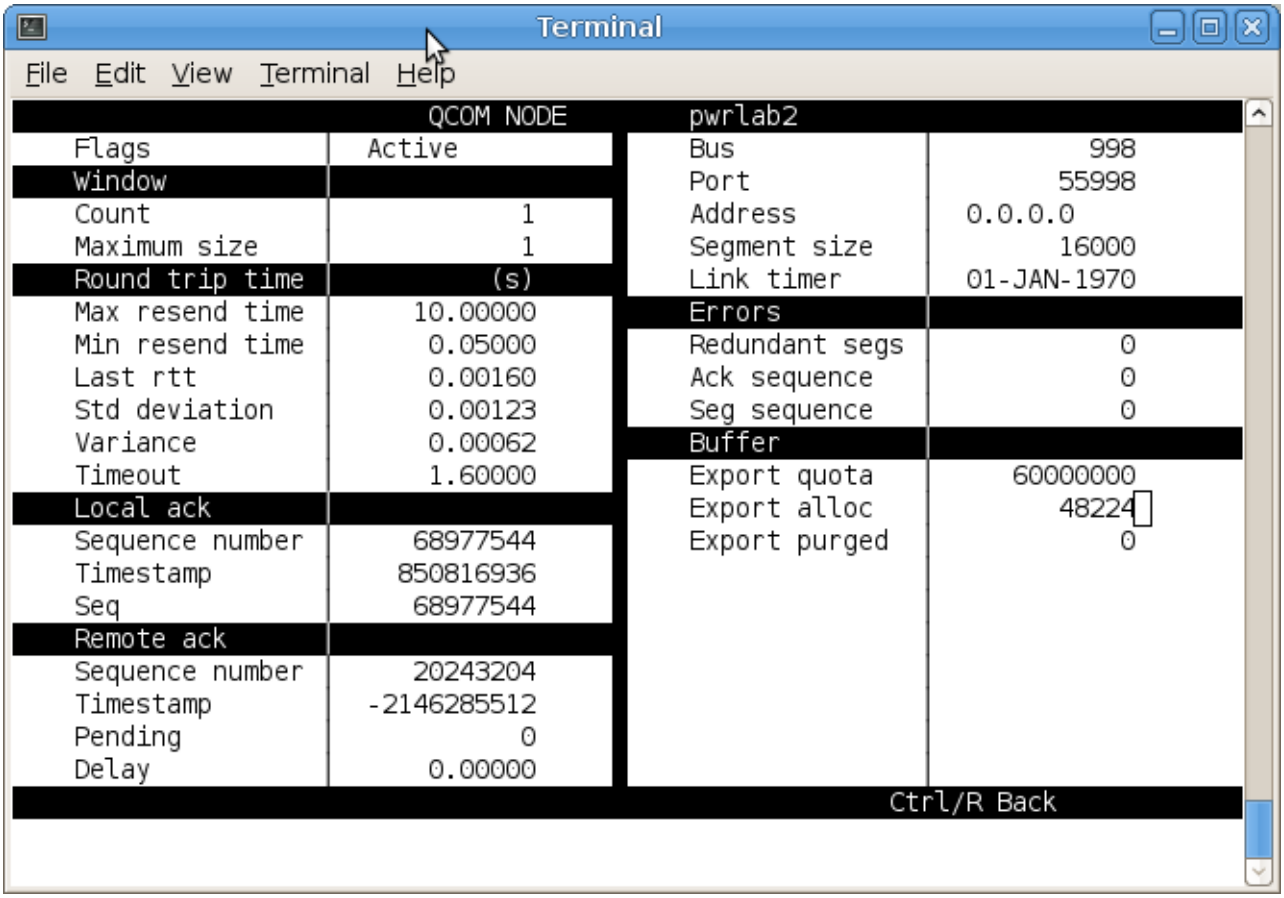


Fig The QCom node picture in rt_rtt

Event handler alarm status message

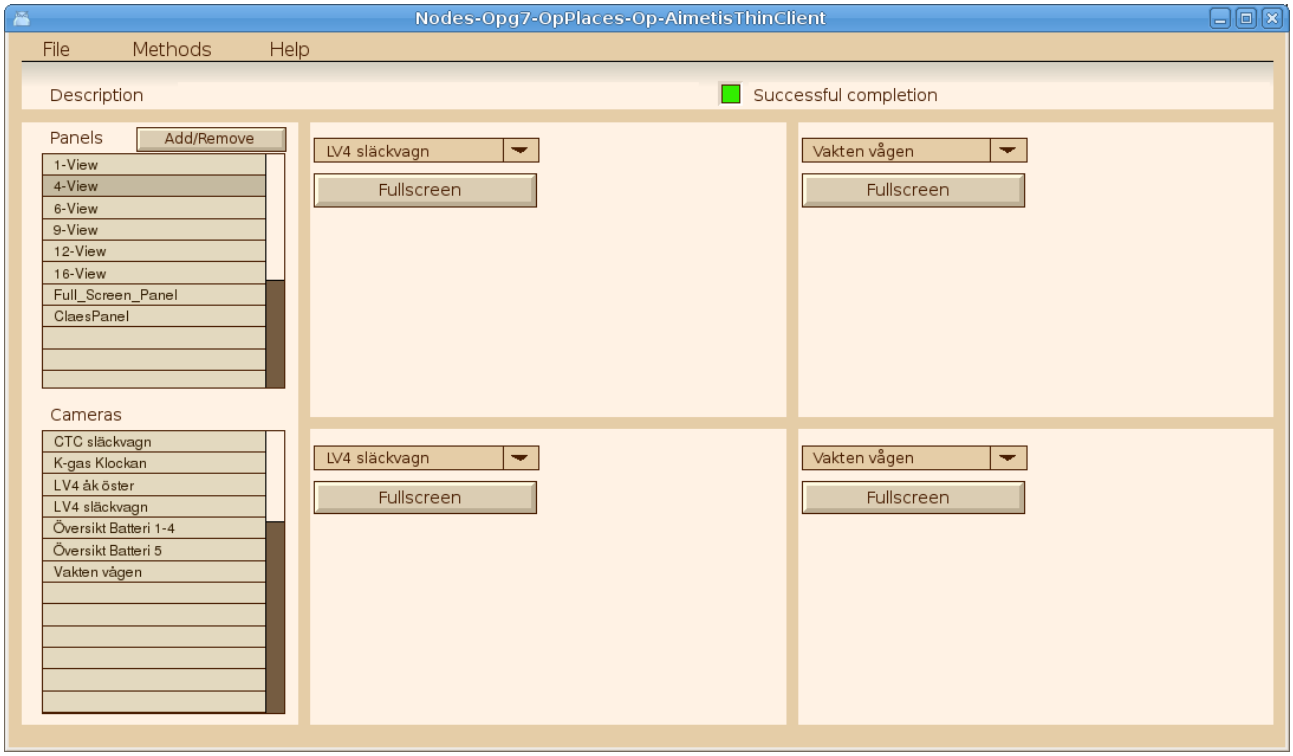
A new cyclic message is introduced in the communication between the event monitor and outunits. The Alarm status message contains a list of all not acknowledged or active alarms. This will ensure that the alarm list always is up to date even at times with bad network when event messages are lost.

Application index

The number of applications that can be supervised is increased from 20 to 50.

Interface to video management system Aimetis

Aimetis Thin Client is a video management system that can display video camera streams, either as panels, ie combinations of different cameras on one screen, or as one cameras full screen. The selection of panels and cameras can be made from Proview with the XttVideoMgm object. Panels or cameras is viewed by selecting a panel or camera in the lists to the left. Panels can also be modified by selecting camera in the in the screen layout to the right. It's also possible to create new panels from the Add/Remove button.



A new server process handles the video communication and this has to be configured with a VideoMgmServer object in the node hierarchy.

Position in Trace and Plc editor navigator window

Sometimes it can be hard to find the rectangle for the currently viewed area in the navigator window in Trace or Plc editor. It's now possible to position the rectangle to the current cursor position with Shift/Click MB1.

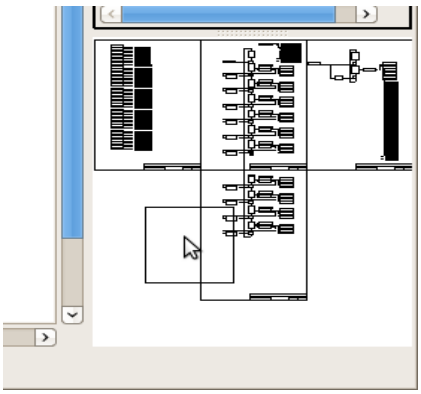


Fig Position the rectangle with Shift/Click MB1

Remote transactions with QCom

It's now possible to send remote transactions on Qcom with the new RemnodeQCom object.

Address calculation for PSS9000 cards

A calculation of the address for PSS9000 card is added. The calculation is made when a card object is created, and is made from the current position in the rack. If the position is changed, and new calculation can be made from the SetAddress method in the popup menu. Also the address switch settings is calculated and displayed in the SwitchSettings attribute.

New Classes

OpPlaceWeb

Configures a web page.

OpPlaceApp

Configures an Android app.

TimeMean

Plc object to calculate a cumulative mean value.

VideoMgmServer

Configuration of the video management server. The video management server handles communication with video management system.

XttVideoMgm

Configuration of the control of a video management system. So far control of Aimetis Thin Client is implemented.

RemnodeQCom

Configuration of remote transactions on QCom.

Modified Classes

WebHandler

Several attributes moved to OpPlaceWeb and OpPlaceApp.

OpPlace

Attribute ColorTheme added to select default color theme.

PID and CompPID

Modification for better handling of long integration times together with short scan times.

NodeConfig and SevNodeConfig

Added attributes for QCom configuration, QComExportBufQuota, QComAckDelay and QComSegmentSize.

FriendNodeConfig

Added attributes for QCom configuration, QComExportBufQuota, QcomAckDelay.

$Node

Arrays for ProcessStatus etc are increased to be able to handle the increased number of application processes.

Ssab_BaseACard, Ssab_BaseDiCard, Ssab_BaseDoCards, Ssab_CO4uP

The attribute SwitchSettings is added to show the desired setting of the address switch on the boards. The attribute VectAddress is removed.

V5.4.1 Additions

Virtual keyboard

The virtual keyboard is used for touch screens and configured in the OpPlace object by settting VirtualKeyboard in options. The virtual keyboard is enabled on input fields in Ge graphs, and for each input field the type of keyboard can be specified in the ValueInput property KeyboardType. The keyboard types are 'standard', 'alphabetic' and 'numeric' displayed in the figures below.

Fig Virtual keyboard of type standard

Fig Keyboard of type alphabetic

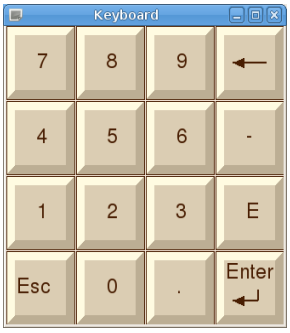


Fig Keyboard of type numeric

Ge dynamics DigScript

Dig script is connect to a digital signal, and executes a script. Dependent on the DigScript.Level property, the script is executed when the signal is high or when the signal goes from low to high.

The script is edited in DigScript.Script.

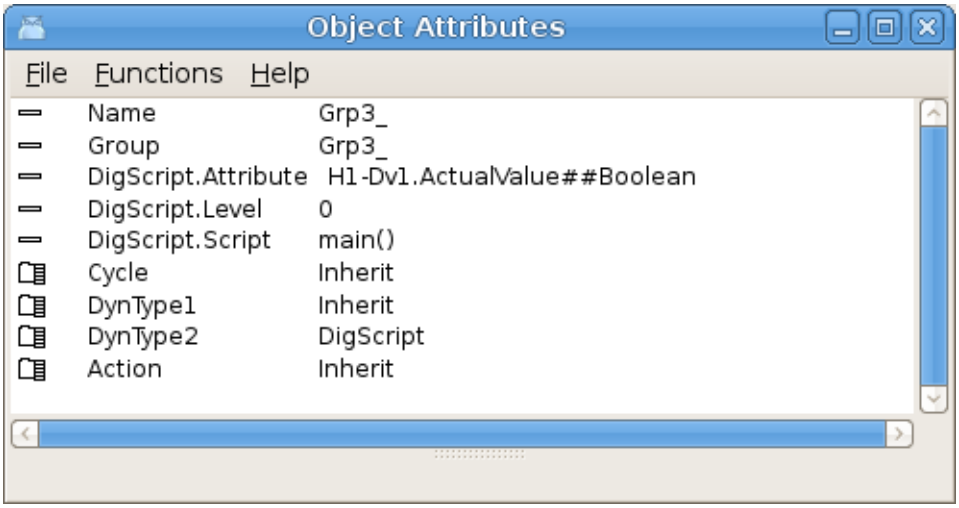


Fig DigScript properties

New property DecimalsAttr for Ge dynamics Value

Ge dynamic Value has new property DecimalsAttr. It is used to state the number of decimals in a float, ie the number of characters after the point. DecimalsAttr should be connected to a signal of type integer or enum that states the number of decimals. A special enum type is created for this, FloatPrecisionEnum that can have the value 0 – 5.

Internal model controller IMC

An internal model controller is implemented in Proview by Bruno PERMANNE, with the objects CompModeIMC and CompIMC. See http://www.proview.se/misc/en_us/CompIMC_doc.pdf for more information.

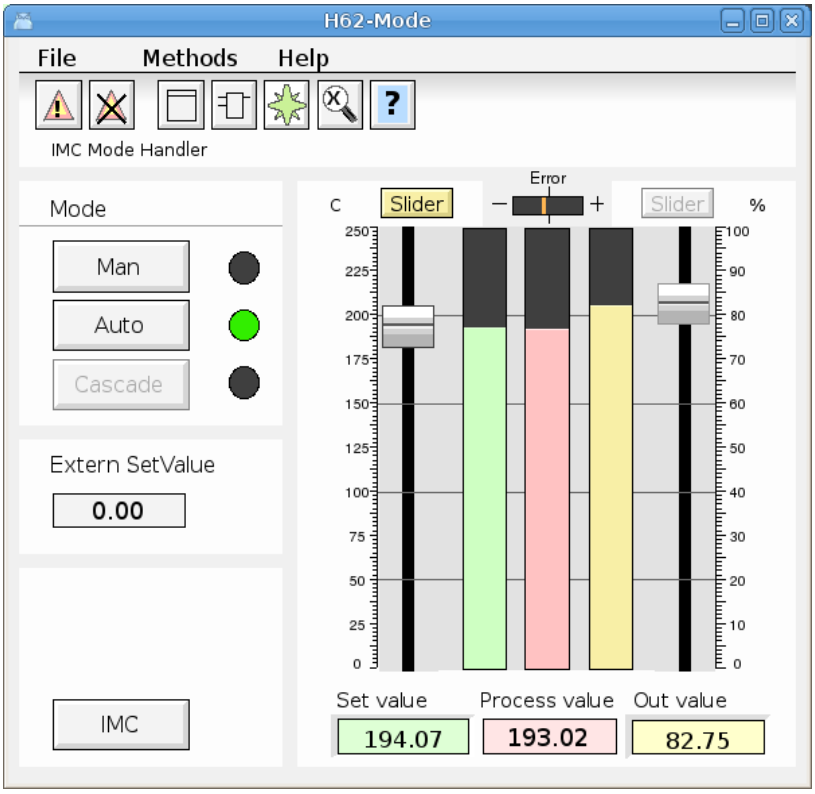


Fig Object graph for mode object CompModeIMC

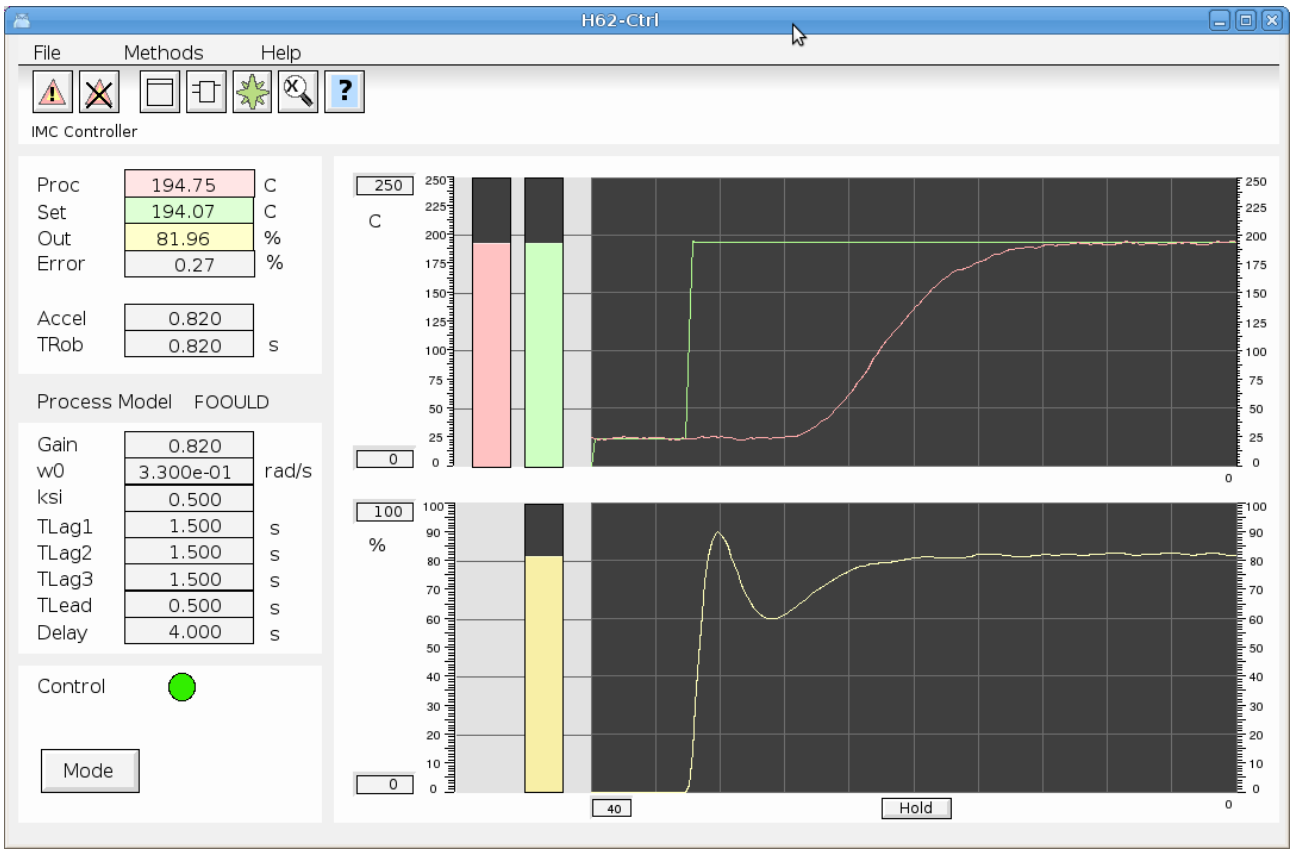


Fig Object graph for controller object CompIMC

Simulate objects

Also some simulate objects are added by Bruno.

Sim_SigGen

This is a signal generator (sine, triangle, rectangle). Noise and offset can be added to simulate real world behavior.

Sim_Integrator

Integrator.

Performs a discrete time integration with dynamic gain.

In Laplace transform:

$$F(s) = \frac{K}{s}$$

Sim_LeadLagFilter

Lead / lag filter.

Performs a discrete time lead / lag filtering.

In Laplace transform:

```
          1 + Tlead.s
F(s) = ---------------
          1 + Tlag.s
```

Sim_LagFilter

Lag filter.

Performs a discrete time lag filtering.

In Laplace transform:

```
                1
F(s) = -------------
          1 + Tlag.s
```

Sim_Delay

Analog 100 cells delay plc function.

Performs a discrete time delay on a 100 float numbers cells shift register.

In Laplace transform:

```
F(s) = exp(-T.s)
```

Sim_SouFilter

Second order underdamped filter.

Performs a discrete time second order underdamped filtering.

In Laplace transform:

```
             1
F(s) = --------------------------------
         1/(w0^2).s^2 + 2.ksi/w0.s + 1
```

Sim_SouTOoFilter

Second order underdamped/overdamped filter.

Performs a discrete time second order uderdamped to overdamped filtering.

In Laplace transform:

```
          1/(w0^2).s^2 + 2.ksi/w0.s + 1
F(s) = ---------------------------------
          (1 + Tlag1.s) . (1 + Tlag2.s)
```

Sim_SlwRateLimiter

Slew rate limiter.

Performs a slew rate limitation.

Upgrade procedure

The upgrading has to be done from any V5.2. If the project has a lower version, the upgrade has to be performed stepwise following the schema

V2.1 -> V2.7b -> V3.3 -> V3.4b -> V4.0.0 -> V4.1.3 ->V4.2.0->V4.5.0->V4.6.0->V4.7.0->V4.8.6->(V5.0.0)->V5.1.0->V5.2.0->V5.3->V5.4

The upgrade procedure is to execute the script upgrade.sh.

If the previous version should be kept, first make a copy of the project.

Make a copy of the project

Do `sdf` to the project and start the administrator

```
> pwra
```

Now the Projectlist is opened. Enter edit mode, login as administrator if you lack access. Find the current project and select Copy Project from the popup menu of the ProjectReg object. Open the copy and assign a suitable project name and path. Save and close the administrator.

Linux release upgrade

If you are using an older Ubuntu version to upgrade the linux release and install the pwr54 package.

Change version

Enter the administrator and change the version of the project to V5.4.0. Save and close the administrator.

upgrade.sh

Do `sdf` to the project.

upgrade.sh is a script that is divided into a number of passes. After each pass you you have to answere whether to continue with the next pass or not.
Start the script with

```
> upgrade.sh
```

Start from the savedirectory pass.

```
Enter start pass [savedirectory] >
```

savedirectory

Save the directory volume.

classvolumes

Create loadfiles and structfiles for the class volumes.

updateclasses

Updates classes and objects to new version.

cnvobjects

Converts or updates specific object.

compile

Compile all the plc programs.

createload

Create loadfiles for the root volumes.

buildnodes

Builds the nodes in the project.

createpackage

Creates distribution packages for the nodes.

List example

```
>
> sdf opg14
Setting base /data0/x5-4-0/rls
>
> upgrade.sh

  upgrade.sh  Upgrade from V5.3.1 to V5.4.0


  Pass

    savedirectory  Save directory volume.
    classvolumes   Create loadfiles for classvolumes.
    updateclasses  Update classes.
    cnvobjects     Create OpPlaceWeb objects.
    compile        Compile all plcprograms in the database
    createload     Create new loadfiles.
    buildnodes     Build all nodes in the project.
    createpackage  Create distribution packages for all nodes in the
project.


-- Upgrade opg14


Enter start pass [savedirectory] >

------------------------------------------------------------------------
 Pass save directory volume
------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Processing line: 51
-- Building volume directory
I Volume directory loaded
```

```
I Database opened /data0/pwrp/opg14/src/db/directory.wb_load
-- Saving file /data0/pwrp/opg14/src/db/directory.wb_load ->
/data0/pwrp/opg14/src/db/directory.wb_load.1
%WNAV-E-MSG, Session saved

Found classvolumes:


------------------------------------------------------------------------------
 Pass create structfiles and loadfiles for classvolumes
------------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y

------------------------------------------------------------------------------
 Pass update classes
------------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Update classes in volume volopg7
Berkeley DB 4.8.30: (April  9, 2010)
info get: 0
W Local class volume "pwrs" [2016-03-09 12:00:00.00]
(/data0/pwrp/opg14/src/db/volopg7.db/pwrs.dbs), in data base "VolOpg7", can
be updated [2016-03-14 11:37:25.53]
W Local class volume "pwrb" [2016-03-09 12:00:00.00]
(/data0/pwrp/opg14/src/db/volopg7.db/pwrb.dbs), in data base "VolOpg7", can
be updated [2016-03-14 11:37:29.63]
...
I A total of 39 objects of 23 classes, and  0 attribute references, were
updated
%WNAV-I-MSG, Classvolumes updated

------------------------------------------------------------------------------
 Pass convert objects in loaded database
------------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Convert volume volopg7
Berkeley DB 4.8.30: (April  9, 2010)
info get: 0
I Database opened /data0/pwrp/opg14/src/db/volopg7.db
--
-- Creating OpPlaceWeb object Noder-opg7-OpWeb
--
%WNAV-E-MSG, Session saved

------------------------------------------------------------------------------
 Pass compile plcprograms
------------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
Berkeley DB 4.8.30: (April  9, 2010)
info get: 0
I Database opened /data0/pwrp/opg14/src/db/volopg7.db
-- Plc window generated            H1-Plc-W
-- Plc window compiled for x86_linux optimized -O3 H1-Plc-W
%GSX-I-CCSUCC, Window compiled           H1-Plc-W
I %GSX-I-CCSUCC, Window compiled   H1-Plc-W
-- Plc plcpgm compiled for x86_linux optimized -O3 H1-Plc

------------------------------------------------------------------------------
 Pass create loadfiles
```

```
------------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
Berkeley DB 4.8.30: (April  9, 2010)
info get: 0
I Database opened /data0/pwrp/opg14/src/db/volopg7.db
-- Building archive for volume: 000_001_001_024
-- Archive built for volume: 000_001_001_024
-- Working with load file volume 'VolOpg7'...
-- Open file...
-- Successfully created load file for volume 'VolOpg7'
-- 40 objects with a total body size of 57256 bytes were written to new
file.

------------------------------------------------------------------------------
 Pass build nodes
------------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Build all nodes
-- Building node aristotle
Berkeley DB 4.8.30: (April  9, 2010)
info get: 0
I Database opened /data0/pwrp/opg14/src/db/volopg7.db
-- Build node aristotle
-- Web startpage generated $pwrp_web/index.html
I Build:    OpPlaceWeb Webpage generated
/data0/pwrp/opg14/bld/common/web/index_opwin_menu.html
I Build:    OpPlaceWeb xtt_help.dat converted to html
I Build:    OpPlaceWeb plc xtthelp-file for volume VolOpg7 converted to html
%WNAV-I-MSG, History html file generated, $pwrp_web/wb_history.html
-- Creating bootfile for node aristotle
-- Plc thread generated priority 22, scantime   0.10000 s, 1 plcpgm's
-- Plc process compiled for x86_linux optimized -O3 Dummy
-- Plc program linked for x86_linux  plc_aristotle_0999_plc
I Build:    Node     aristotle
%LFU-S-SUCCESS, successful completion

------------------------------------------------------------------------------
 Pass create packages
------------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Creating distribution packages for all nodes
W No file found: $pwrp_load/rtt_crr_000_001_001_024.dat
W No file found: $pwrp_load/rtt_crro_000_001_001_024.dat
W No file found: $pwrp_load/rtt_crrc_000_001_001_024.dat
W No file found: $pwrp_load/rtt_crrs_000_001_001_024.dat
W No file found: $pwrp_load/pwrp_alias.dat
W No file found: $pwrp_exe/*.pwg
W No file found: $pwrp_load/aristotle/b55/Rt_xtt:
$pwrp_load/aristotle/Rt_xtt:$pwrp_load/Rt_xtt
W No file found: $pwrp_load/aristotle/b55/xtt_setup.rtt_com:
$pwrp_load/aristotle/xtt_setup.rtt_c
W Distribute warnings node aristotle: 8 warnings
I Distribute package for node aristotle

-- The upgrade procedure is now accomplished.
```