



Release Notes V5.0

2013 10 23

Copyright SSAB Oxelösund AB 2013

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

Table of Contents

Upgrading to Proview V5.0.0.....	4
New functions.....	4
Ethernet Powerlink.....	4
Proview as MN.....	4
Proview as CN.....	4
EplHandler.....	4
Multiview operator interface.....	4
Alarmview and multiple alarm lists.....	6
Android application.....	7
Graphs opened from pwg files in web interface.....	9
Object graphs in web interface.....	9
Building development environment in ARM.....	9
Wb history in html.....	9
Export of History curves from curve window.....	9
Show existence of MoreText in alarm and event list.....	10
Show backup file content.....	10
Eventlog statistics.....	10
Ge scale snap to grid.....	11
Loadfile version check in build node.....	11
Component for frequency converter ABB ACS880	11
Ge graph always on top.....	11
New Classes.....	12
EplHandler.....	12
EplMN.....	12
EplCN.....	12
Epl_Module.....	12
Epl_CNServer.....	12
Epl_CNServerModule.....	12
BR_Di4_EplModule.....	12
BR_Do4_EplModule.....	12
BR_Ai4_EplModule.....	12
BR_Ao4_EplModule.....	12
XttMultiView.....	12
AppGraph.....	13
Modified Classes.....	13
Upgrade from V4.8.X to V5.0.0.....	13

Upgrading to Proview V5.0.0

This document describes new functions in Proview V5.0.0, and how to upgrade a project from V4.8 to V5.0.0.

New functions

Ethernet Powerlink

Powerlink, an ethernet fieldbus based on the CAN protocol, is now added to the Proview IO system. A Powerlink network consists of one Managing Node (MN) and a number of Controlled Nodes (CN). Both the MN and CN are implemented in Proview which thus can act as both master or device. The Powerlink network is configured in the openCONFIGURATOR, and a corresponding configuration also has to be done in the Proview database, with Agent, Rack, Card and channel objects.

Proview as MN

When Proview acts as a Managing Node, the MN is configured with a Epl_MN in the node hierarchy. Below this, one Epl_CN object is created for each device on the network. A CN contains modules, configured with CN_Module object. There are also a number of CN_Module objects available for B&R modules and ACS880.

The openCONFIGURATOR is opened from the popup menu of a Epl_MN object.

Proview as CN

When Proview acts as a Controlled Node, it is configured with a Epl_CNServer object, and below this Epl_CNServerModule objects.

EplHandler

The Powerlink stack is executed in a separate server process configured with a EplHandler object. The Process attribute should be set to Powerlink for all Powerlink IO objects. For this reason the Powerlink IO will not be synchronized with the Plc program. The Powerlink stack will not function when it's started by a plc thread, but hopefully this problem can be solved in the future.

See also Guide to IO System for more information about the Powerlink configuration.

Multiview operator interface

Multiview is a window organized as table where each cell can contain a graph, a trend, a sev history curves, an alarm list or another multiview window. The example below displays a multiview window with one column and two rows. The first cell contains another multiview with two columns the tree rows, and the second cell contains an alarm window. The different alarm windows show alarms from different parts of the plant, specified by alarmviews.

An multiview window is configured by a XttMultiView object. The Action array contains the specification for each cell.

It is also possible to exchange a graph or curve in a cell with the 'set subwindow' command.

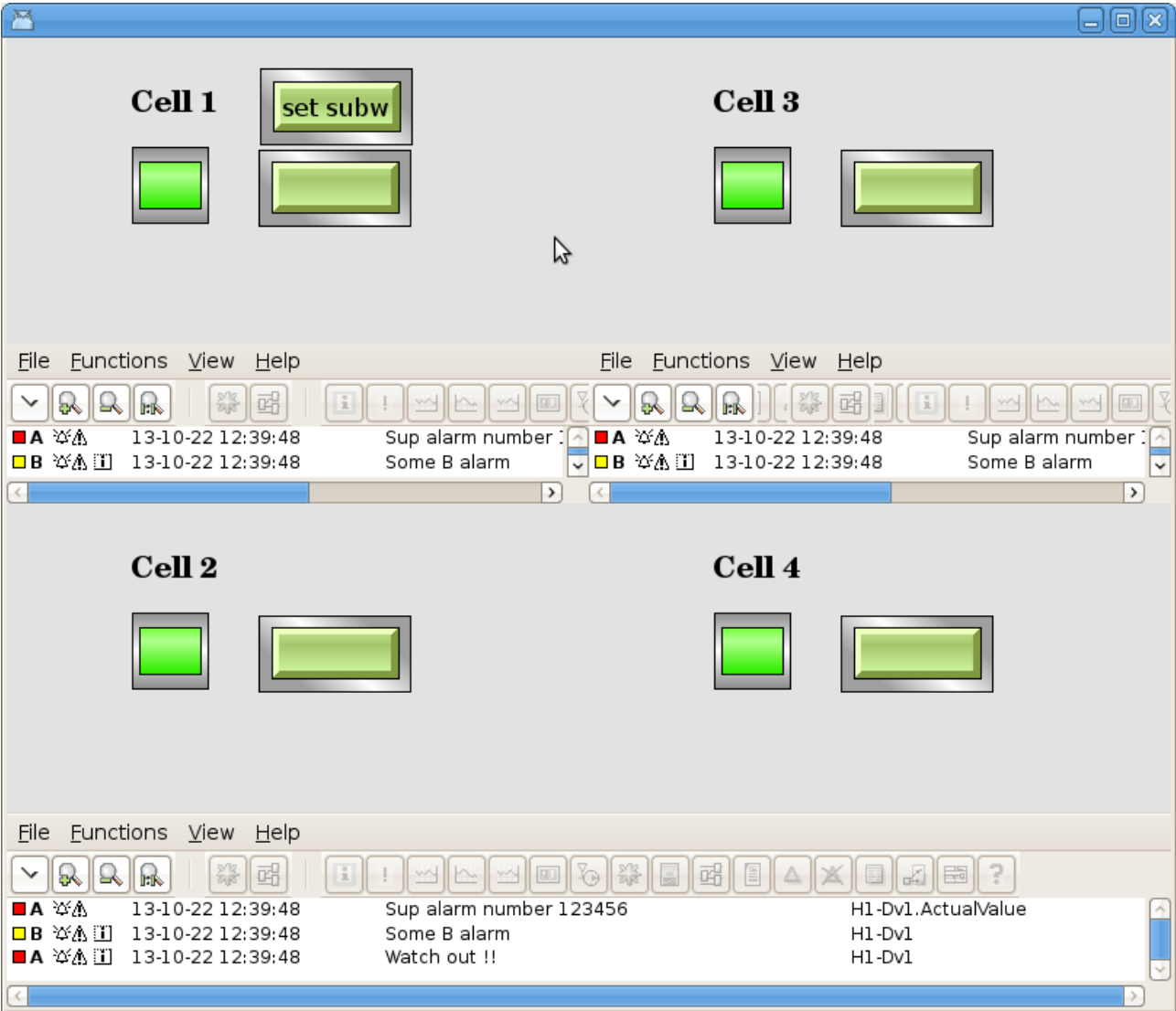


Fig Multiview window

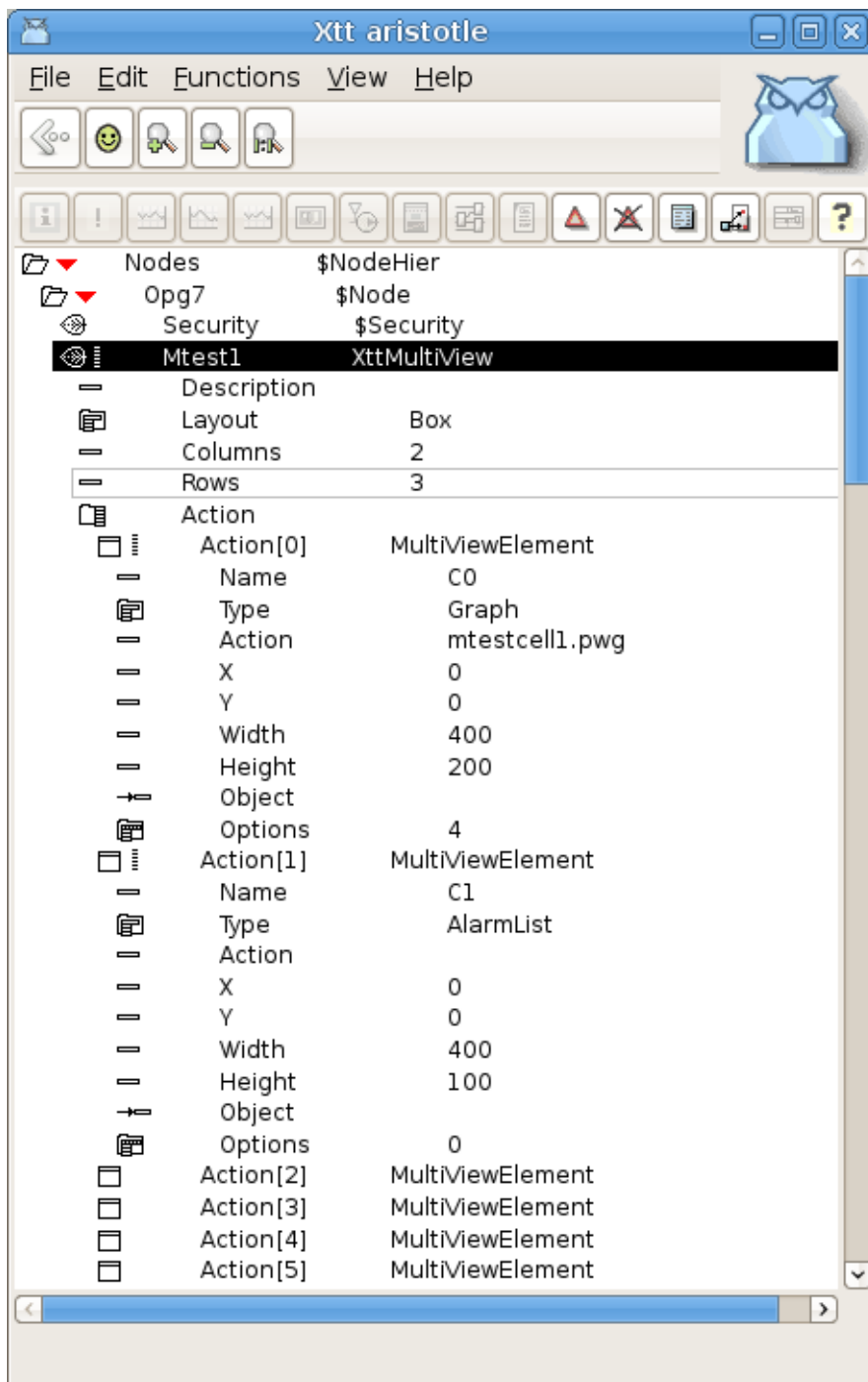


Fig Multiview configuration

Alarmview and multiple alarm lists

It is possible to open a number of different alarm lists. The alarms displayed in the different alarm lists are specified in AlarmView objects. The alarms can be categorized and placed in to different maps.

An alarm view is configured with an AlarmView object. Every map in the alarm view is configured by an AlarmCategory object that contains an array of hierarchies. Alarms from these hierarchies will be displayed in this map.

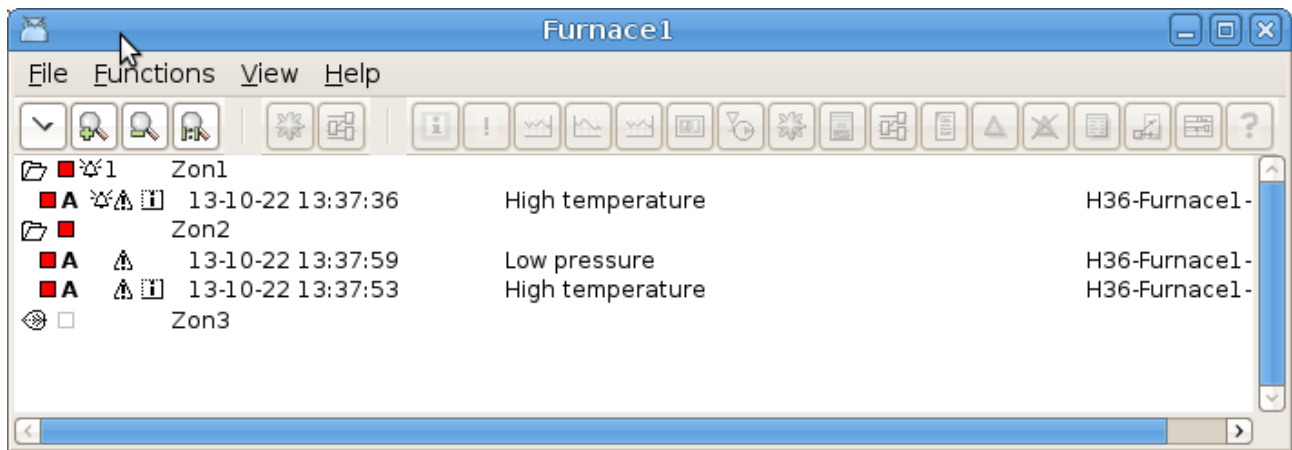


Fig AlarmView for furnace 1

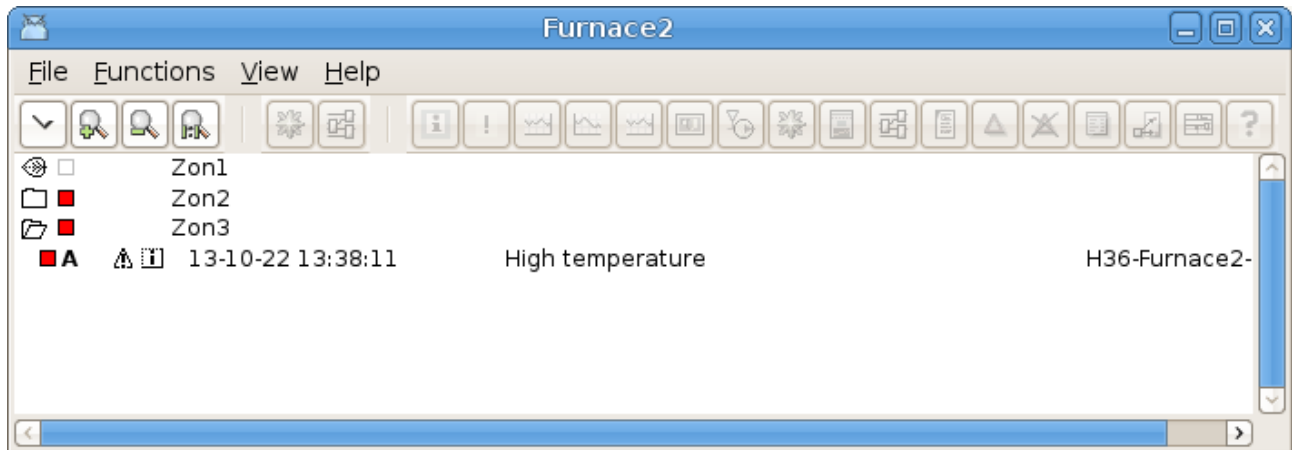


Fig AlarmView for furnace 2

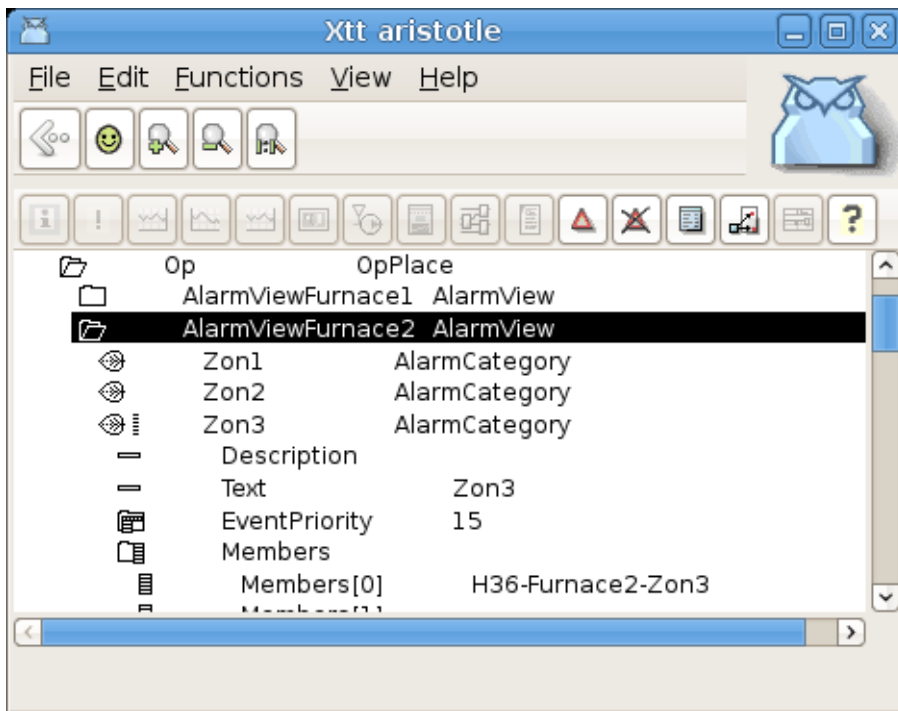


Fig configuration of alarm view

Android application

V5.0.0 contains a beta version of an android application. The application uses the same communication protocol as the web interface in Proview and uses the same server processes. These are, as before, configured with the WebHandler object. A new object is the AppGraph object, that can be positioned below the WebHandler object to specify Ge graphs for the application.

The app contains the the following functionality

- Access control with login and user privileges.
- Runtime navigator.
- Plc trace.
- Alarm list.
- Event list.
- Display of cross references for signals.
- Ge graphs.

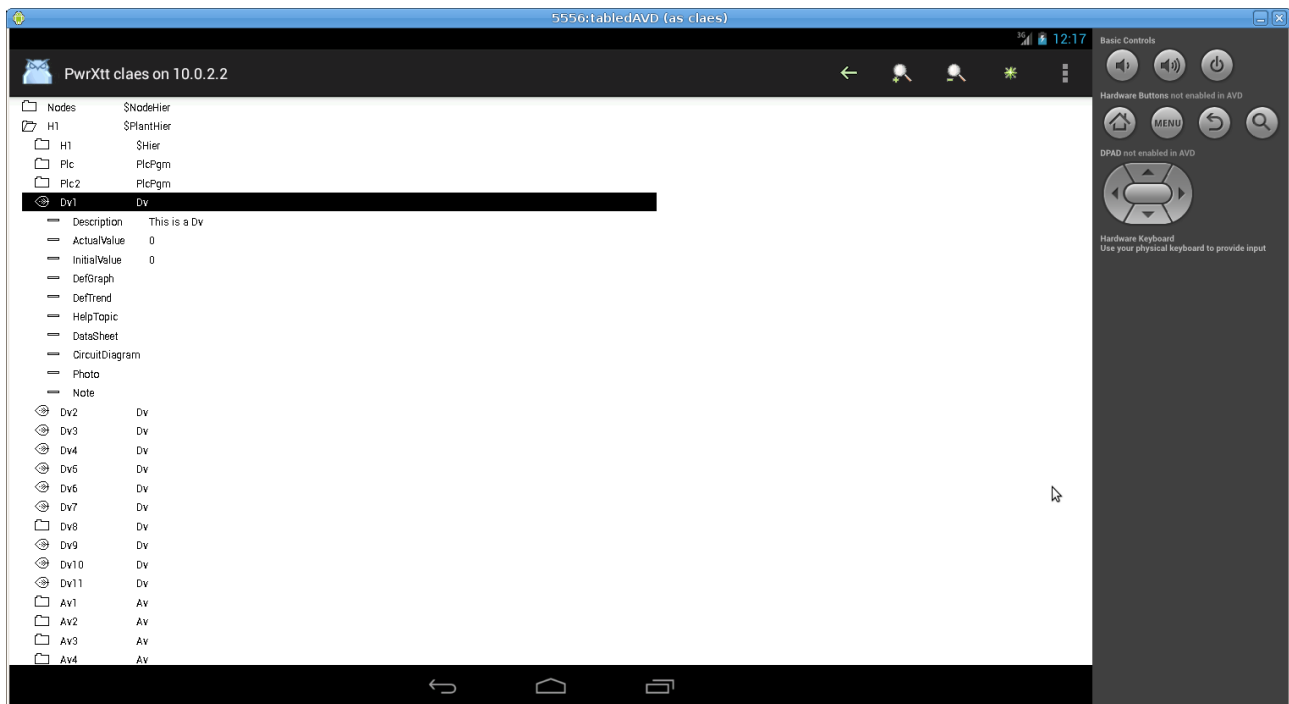


Fig Navigator in android application

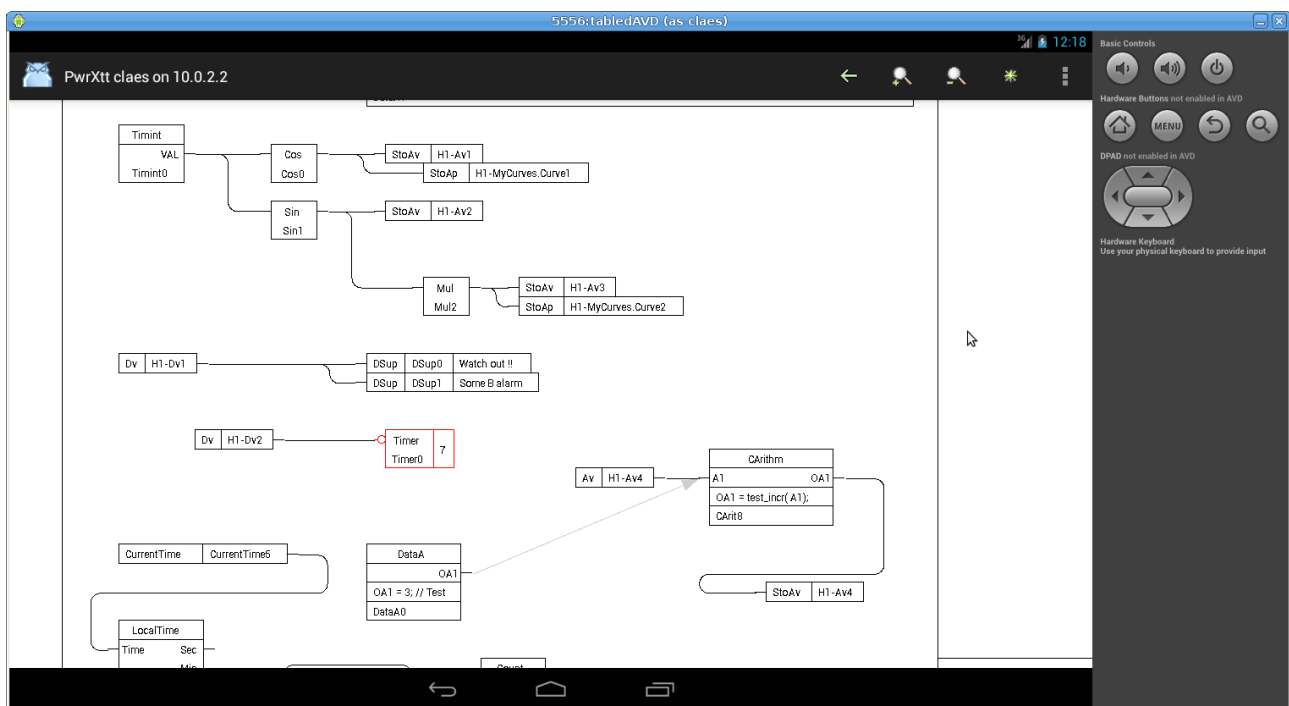


Fig Plc trace in android application

Graphs opened from pwg files in web interface

Previously when a graph was displayed in the web or java interface, it had to be exported as a java frame or a java applet. Exporting to java could be a problem for large graphs as there is a limitation in size of java functions, and large jar-files could also result in slow performance. Now the .pwg file is read directly from the \$pwrp_web catalog instead.

Object graphs in web interface

Object graphs for BaseComponent objects wasn't available in the web interface before, as the jar archive was too large to get acceptable performance. Now as the pwg-files are used instead, these object graphs are available.

Building development environment in ARM

It is now possible to build the development package on ARM.

Wb history in html

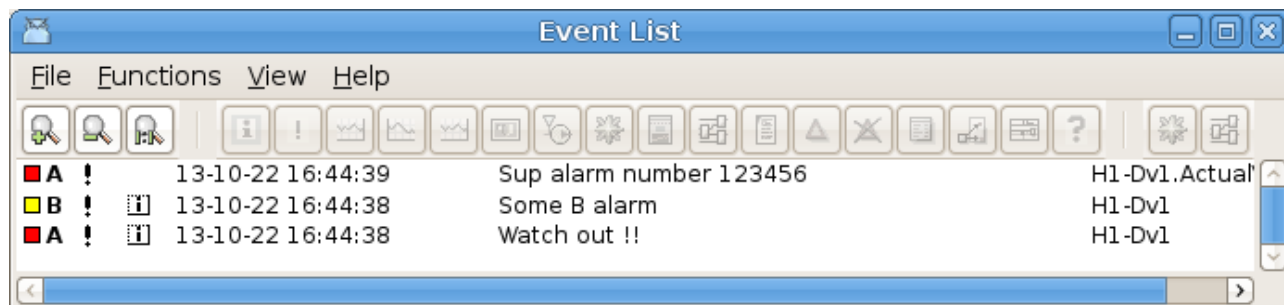
The wb history is converted to html on \$pwrp_web and can be opened with a link to wb_history.html. Note! If the comment is enabled in the configurator (Options/Settings/Hierarchy) this page will be more interesting.

Export of History curves from curve window

The export function in the history curve window, File/Export in the menu, has the additional choice "All attributes". All curves will be exported to text file, and this file can later be imported with the xtt command "open history/file="

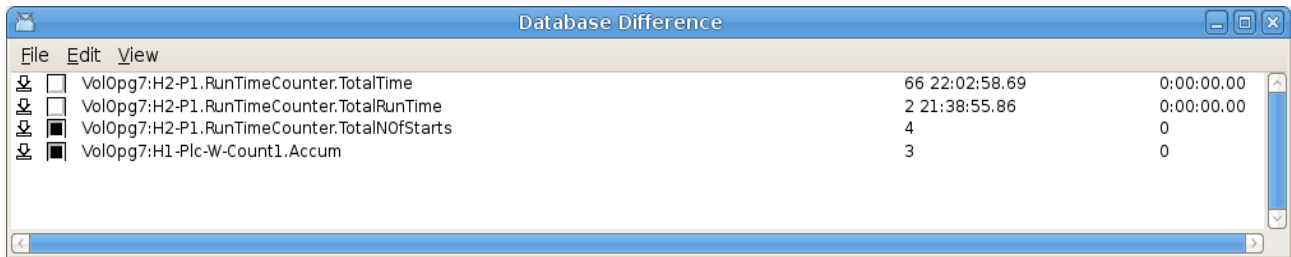
Show existence of MoreText in alarm and event list

If there is a MoreText for an alarm or event, this is marked with an info icon in the alarm and event list. When the info icon is clicked, the MoreText is displayed.



Show backup file content

The configurator command "show backup" will open the backup utility, that can show the content of a backup file. It is also possible to compare two backup files, or to compare the values in the backup file with the values in the development database. Marked values can then be stored to the database.



<input type="checkbox"/>	VolOpg7:H2-P1.RunTimeCounter.TotalTime	66 22:02:58.69	0:00:00.00
<input type="checkbox"/>	VolOpg7:H2-P1.RunTimeCounter.TotalRunTime	2 21:38:55.86	0:00:00.00
<input checked="" type="checkbox"/>	VolOpg7:H2-P1.RunTimeCounter.TotalNOFStarts	4	0
<input checked="" type="checkbox"/>	VolOpg7:H1-Plc-W-Count1.Accum	3	0

Fig Found differences between the values in the backup file and the development database

Eventlog statistics

Event statistics, ie the number of times and event is registred in the evenlog database, can be displayed from File/Statistics in the Eventlog dialog.

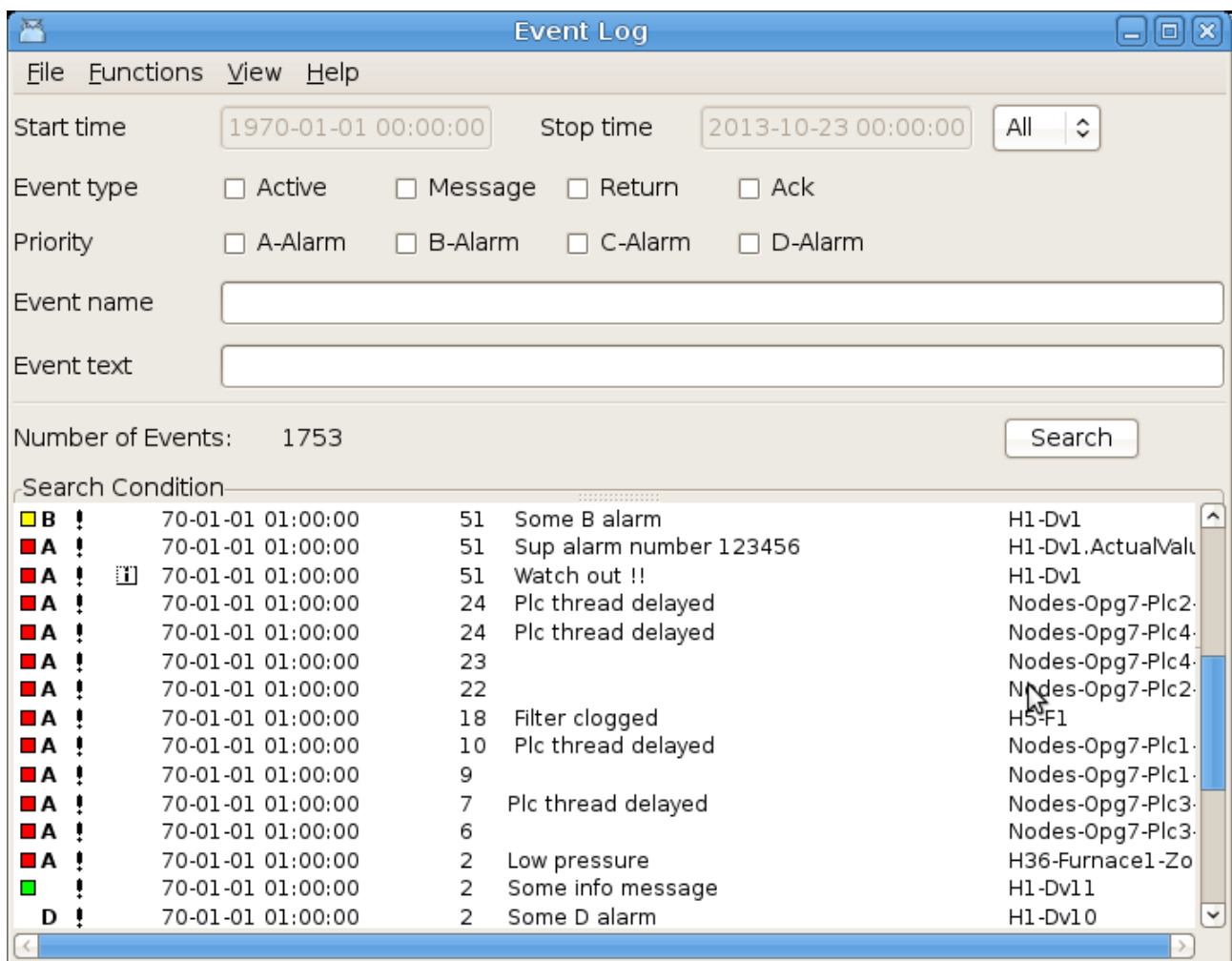


Fig event log statistics

Ge scale snap to grid

Snap to grid for the Ge scale function implemented.

Loadfile version check in build node

A version check of load-files is made when the node is built in addition to the check that is made when the distribution package is built.

Component for frequency converter ABB ACS880

New component objects for frequency converter ABB ACS800 with Profibus or Powerlink communication.

Ge graph always on top

It is possible to open a graph that is always on top of another graph. This is done with the /pwindow and /pinstance qualifiers to the 'open graph' command. /pwindow specifies the name of the parent graph, and /pinstance instance object for the parent graph, if parent graph is an object graph.

New Classes

EplHandler

Object to configure the Powerlink server.

EplMN

I/O Agent object configuring a Powerlink Managing Node.

EplCN

I/O Rack object configuring a Powerlink Controlled Node.

Epl_Module

I/O Card object configuring a Powerlink module.

Epl_CNServer

Object configuring the Proview node as a Powerlink Controlled Node.

Epl_CNServerModule

Object configuring a module in a Powerlink server.

BR_Di4_EplModule

Power link module object for a B&R device with 4 Di channels.

BR_Do4_EplModule

Power link module object for a B&R device with 4 Do channels.

BR_Ai4_EplModule

Power link module object for a B&R device with 4 Ai channels.

BR_Ao4_EplModule

Power link module object for a B&R device with 4 Ao channels.

XttMultiView

Configures a multiview window in a operator station.

AppGraph

Configures a Ge graph in an android app.

Modified Classes

Upgrade from V4.8.X to V5.0.0

Enter the administrator and change the version of the project to V5.0.0. Save and close the administrator.

If you have any class volumes, enter the class editor and build the volume.

Enter the configurator for each root volume and activate 'Function/Update Classes' and build.