Release Notes V5.5

2017 03 07

Table of Contents

Upgrading to Proview V5.5.0

This document describes new functions i Proview V5.5.0, and how to upgrade a project from V5.4.0 to V5.5.0.

New functions

Horizontal and vertical marker added to XY_Curve and Trend

For the Ge trend component, two horizontal marker lines are added. The position of the lines are controlled by the Trend properties Trend.Mark1Attr and Trend.Mark2Attr that can be connected to analog signals. The color of the marker lines is set with Trend.Mark1Color and Trend.Mark2Color.

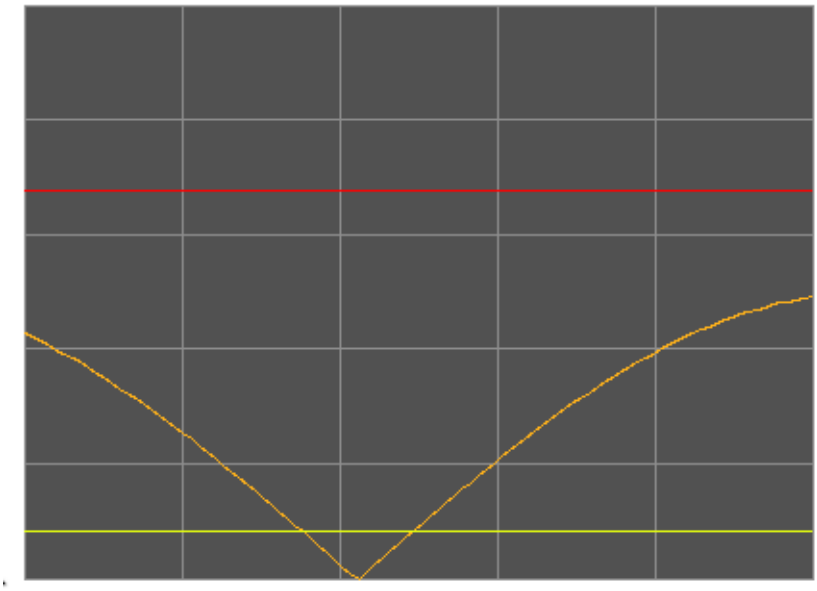


Fig Ge trend curve with red and yellow marker lines

Also the XY_Curve has new marker lines, two horizontal and two vertical. The horizontal maker lines are controlled by XY_Curve.YMark1Attr and XY_Curve.YMark2Attr, and the vertical by XY_Curve.XMark1Attr and XYCurve.XMark2Attr. The colors of the marker lines can be set with XY_Curve.Mark1Color and XYCurve.Mark2Color.

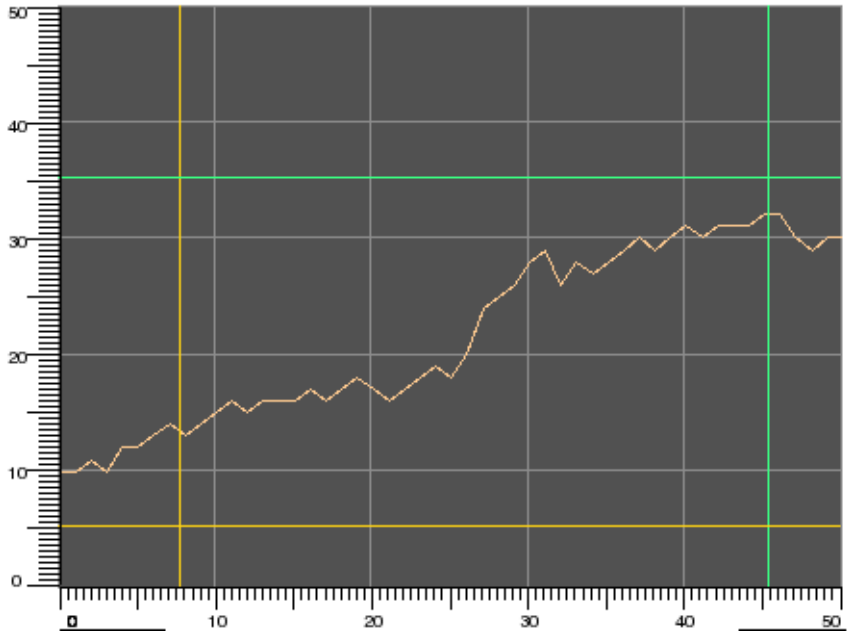


Fig XY_Curve with two horizontal and two vertical maker lines

Table/Curve component

A new table/curve object, CompCurveTab is available. The object contains a curve specified with x and y values for 50 points.

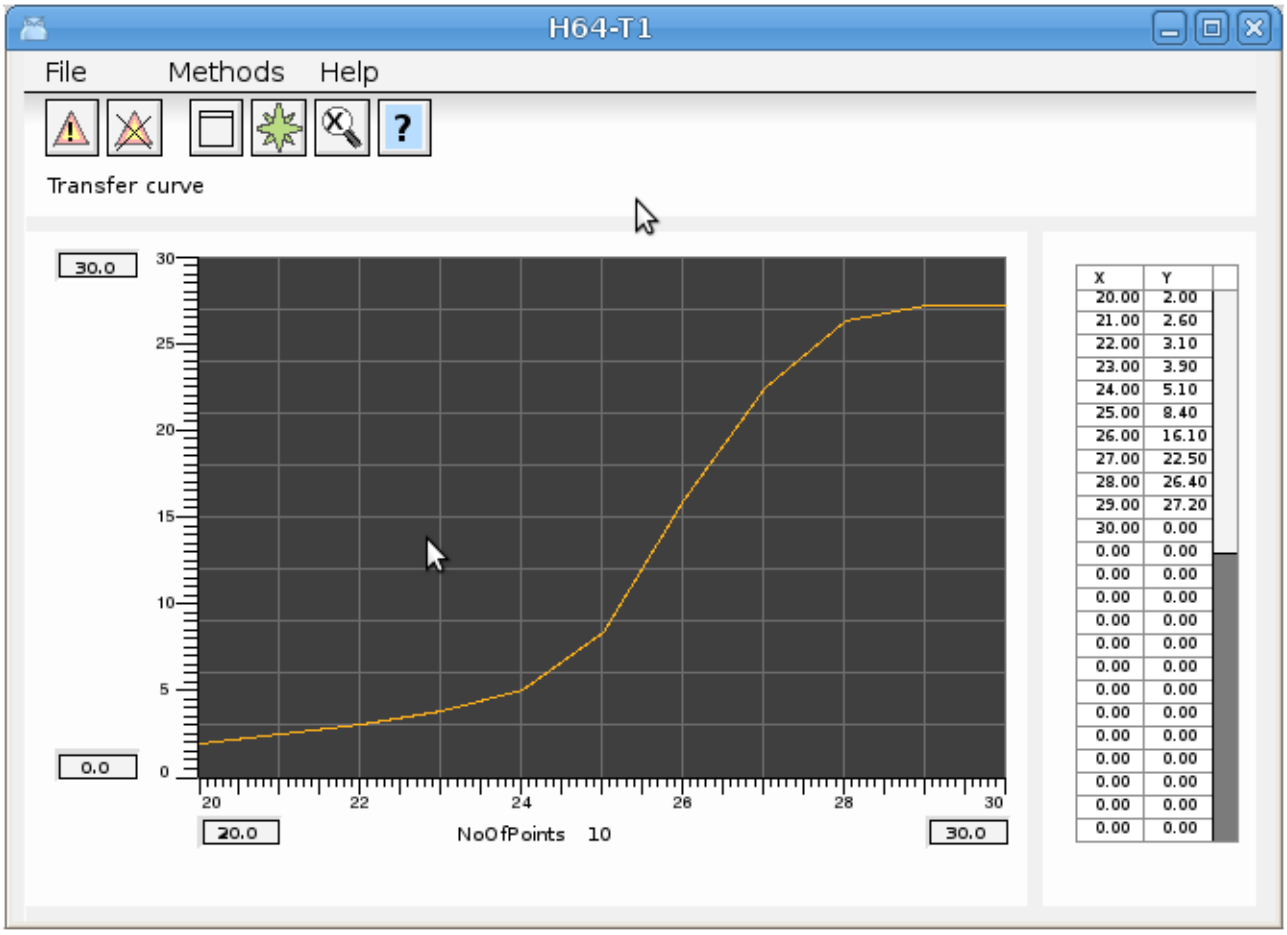


Fig Object graph for the CompCurveTab displaying the curve and table

The CompCurveTabValue object is used get the y value for a specific x value of the curve. Several

value objects can be connected to the same curve object. The CompCurveTabValue object has a function object that should be inserted in the plc code, CompCurveTabVauleFo.

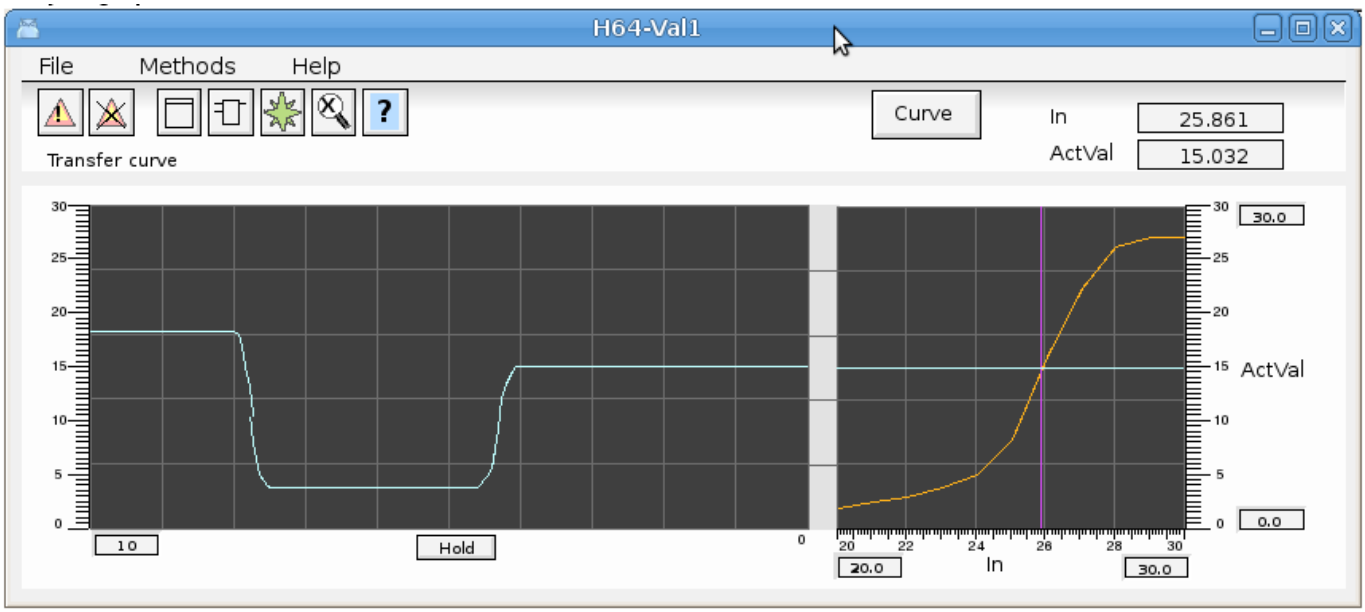


Fig Object graph for CompCurveTabValue

Polynomial curve component

CompCurvePol is similar to the CompCurveTabl object, but the curve is specified with a polynom instead. The CompCurvePolValue is used to fetch the y value for a specific x value from the curve.

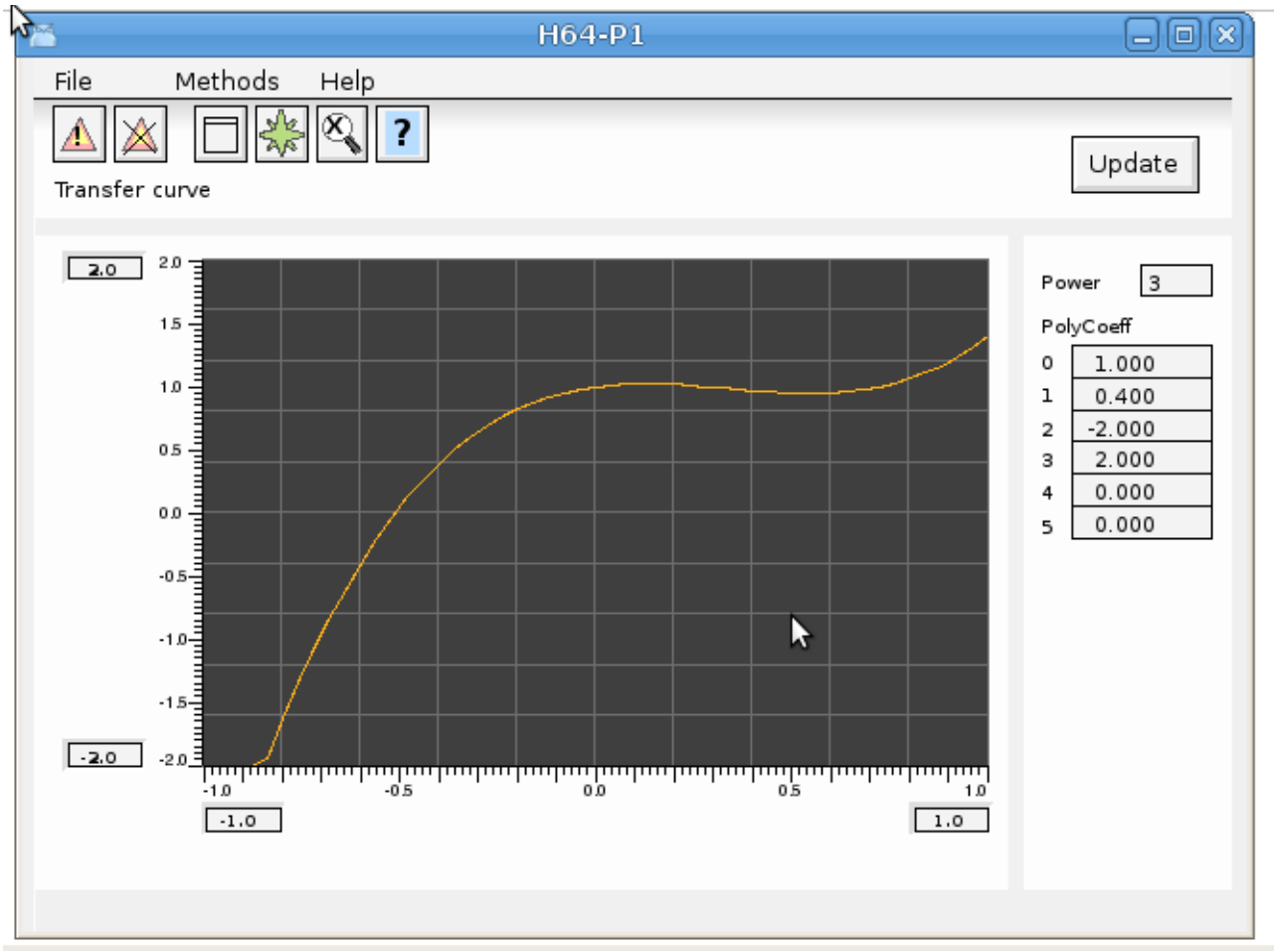


Fig Object graph for CompCurvePol

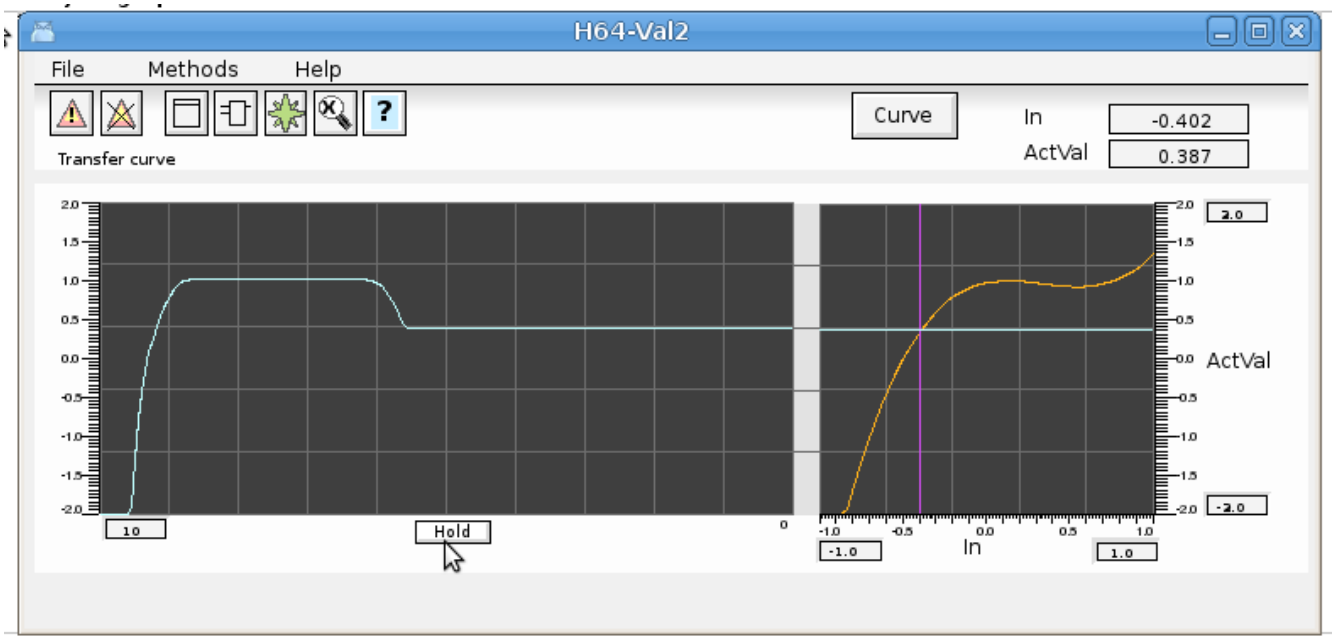


Fig Object graph for CompCurvePolValue

Xtt tree window to view specific classes

A new tree window that shows all objects of a specific class in the current node and in mounted nodes. It is for example used in the operator window for trends and fast curves to show curves also in mounted nodes. Previously only curves in the current node was displayed.

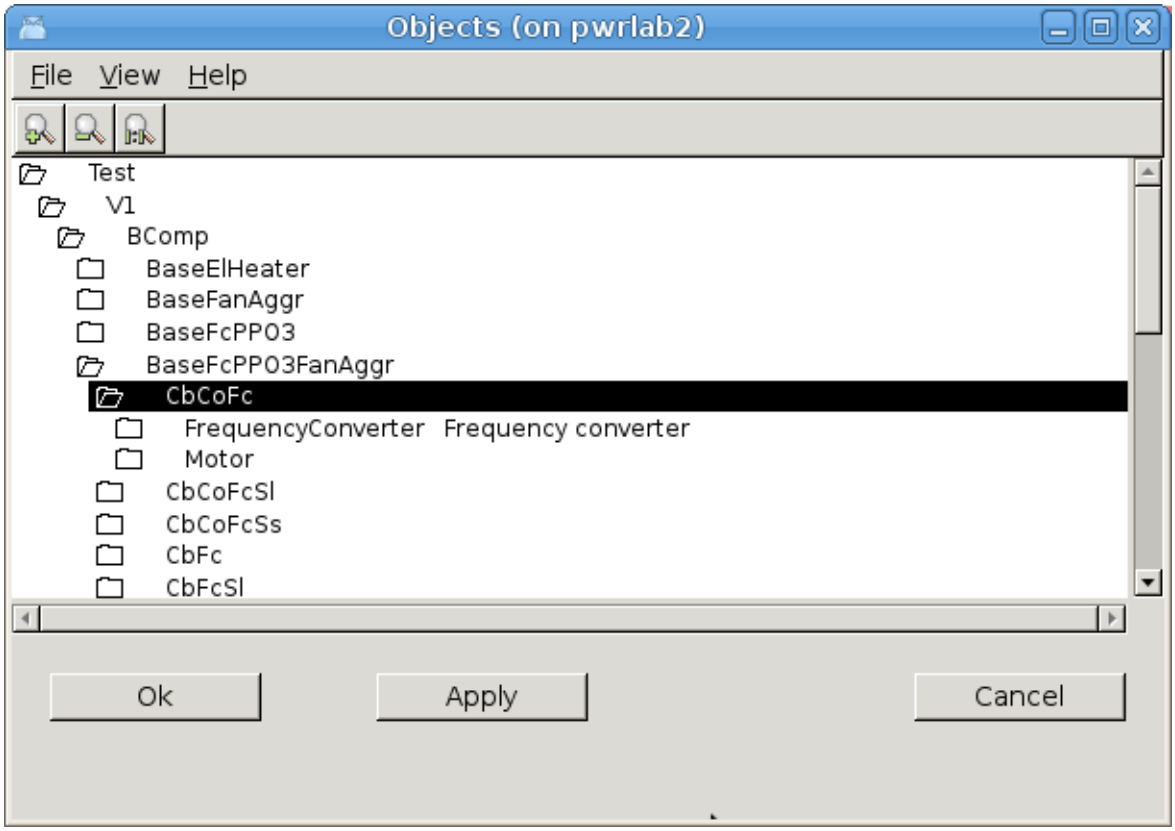


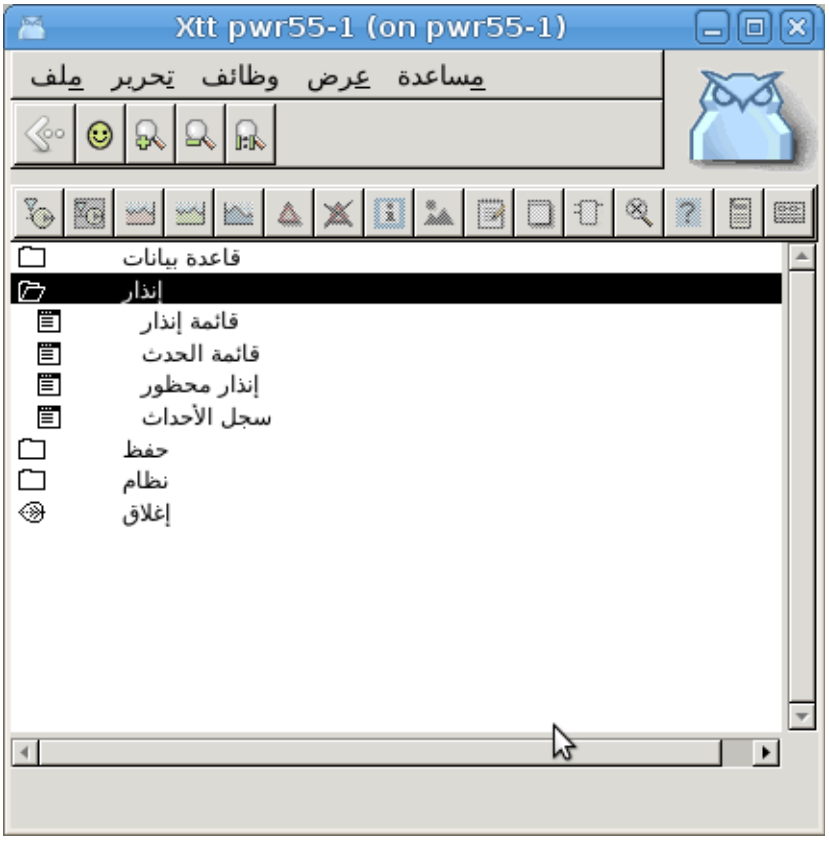
Fig Tree window for Trend curves

The tree window is opened with the xtt command 'show objecttree /class=', for example when displaying the trend objects 'show objecttree /class=dstrend,dstrendcurve'.

A new nethandler request is added to fetch all objects of a specific class on a remote node. Thus the object rree window will not work with mounted nodes of older versions.

Xtt translation to syrian arabic

An xtt translation is made to syrian arabic. To open the syrian arabic version start rt_xtt with option -l ar_sy.



SuppressSup, suppress alarms

SuppressSup is a function object to suppress alarms. The object is connected to a DSup, ASup, DSupComp or ASupComp object, and will suppress the sup object if the input is high. The object has an output that is high if the sup object is suppressed or if the sup object is active. In this way the output can be connected to SuppressSup objects to suppress alarms on a lower level.

Wb script fuction InLib()

The wb script function InLib() tests if an object is in a library hierarchy or not.

int inLib(string name)

Example

```
if ( InLib( "H1-Motor"))
  ...
endif
```

Xtt script functions GetGraphInstance() and GetGraphInstanceNext()

The Xtt script functions GetGraphInstance() and GetGraphInstanceNext() makes is possible to loop through all the currently open instances of an object graph.

Ge text object size

The text object in Ge previously had a fix size of 80 characters. This is now changed to dynamic size.

Ge dynamic ValueInput.UpdateOpen

UpdateOpen is added to the ValueInput dynamics. The displayed value in the field will be updated also when the field is open for input.

Ge signals

Ge signal can be used to trigger actions in Ge Graphs. A signal is emitted with the action EmitSignal that will emit a signal when an object is clicked on. A signal can also be emitted with the xtt command 'emit signal' where the signal can be directed to a specific graph or to all graphs.

The signal is caught with the Ge action CatchSignal, that will execute all the actions on the current object when the signal is caught.

Signals can for example be used in function keys to define different functions in different graphs.

Sev deadband with linear regression

A new type of two dimensional deadband is added to the Sev functionality. The deadband is configured in SevHist.Options with the DeadBandLinearRegr option.

The previous deadband only calculated vertical differencies while DeadbandLinearRegr calculattes a straight line of arbitrary direction with linear regression. A new value is only stored if distance from the line to any point is larger than the deadband. This will in many cases drastically reduce the number or stored points and improve the storage capacity. In the figure below only the black points will be stored.

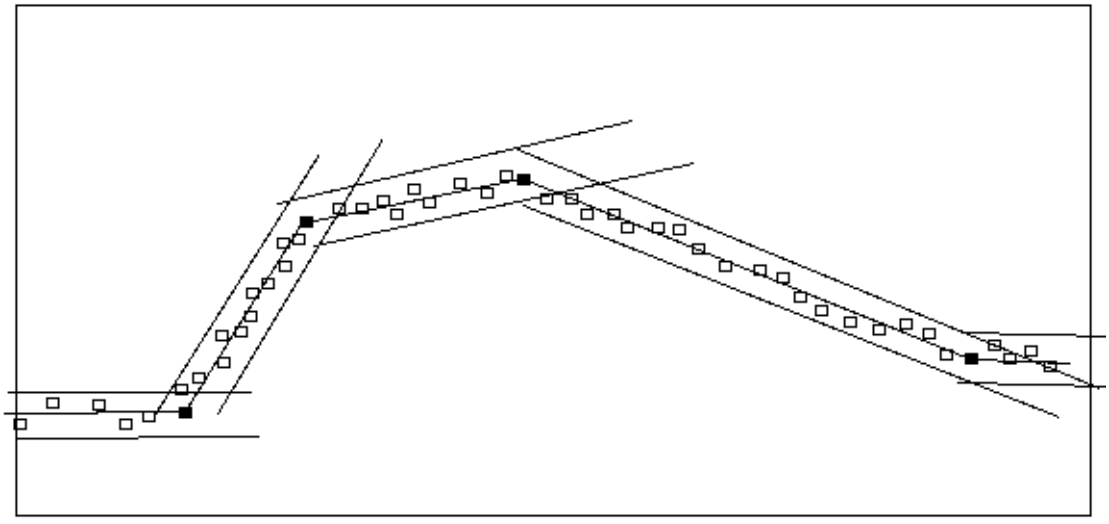


Fig Storage with deadband linear regression

Redundancy

A beta version of redundancy is implemented in V5.5 with PSS9000 I/O.

wbl parser antlr replaced

Antlr was previously used to parse wb_load and wb_dmp files. This is now replaced by an parser developed in the Proview project. Thus the dependency on antrl when building from sources are removed.

Build environment dependencies for wbl-files and exec-files

New dependencies for wbl-files and exec-files are added when building Proview from sources.

New Classes

SuppressSup

Suppress alarms..

Modified Classes

CompModePID and CompPID

Description attribute added to CompModePID and CompPID.

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->V5.5

The upgrade procedure is to dump the database with reload.sh, change the version of the project in the projectlist, and then execute the script upgrade.sh.

NOTE !!

Do not activate Update Classes.

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.

Dump the databases

Execute the first pass, *dumpdb*, in the script *`reload.sh`*.

```
> reload.sh

  reload.sh   Dump and reload of database.

  Arguments   Database or databases to reload.
              I no arguments is supplied, all databases will be
              reloaded.

  Pass

    dumpdb       Dump database to textfile $pwrp_db/'volume'.wb_dmp
    classvolumes Create structfiles and loadfiles for classvolumes
    renamedb     Rename the old database
    loaddb       Load the dump into the new database
    compile      Compile all plcprograms in the database
    createload   Create new loadfiles.
    createboot   Create bootfiles for all nodes in the project.


-- Reloading volume  directory volopg2

Pass: dumpdb classvolumes renamedbloaddb compile createload createboot

Enter start pass [dumpdb] >

------------------------------------------------------------------------
 Pass dump database
------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
ls: cannot access /data0/pwrp/opg2/common/db/*.wb_dmp: No such file or
directory
Dumping volume directory in /data0/pwrp/opg2/common/db/directory.wb_dmp
...
I Database opened /data0/pwrp/opg2/common/db/volopg2.db
ls: cannot access /data0/pwrp/opg2/common/db/*.wb_load: No such file or
directory



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

Check that the one dumpfile is created for every rootvolume

```
> cd $pwrp_db
> ls -l *.wb_dmp
-rw-rw-r-- 1 cs pwrp 7467 2010-03-26 16:32 volopg2.wb_dmp
```

Linux release upgrade

If you are using an older Ubuntu version upgrade the linux release and install the pwr55 package.

Change version

Enter the administrator and change the version of the project to V5.5.0. 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 classvolumes pass.

```
Enter start pass [classvolumes] >
```

classvolumes

Create loadfiles and structfiles for the class volumes.

renamedb

Store the old databases under the name `$pwrp_db/'volumename'.db.1.`

loaddb

Create databases and load the dumpfiles into them.

compile

Compile all the plc programs.

createload

Create loadfiles for the root volumes.

createboot

Create bootfiles for all nodes in the project.

If the project contains any application programs, these has to be built manually.

Delete files from the upgrading procedure:

`$pwrp_db/*.wb_dmp.*`
`$pwrp_db/*.db.1` (old databases, directories which content also should be removed)

List example

```
>
> sdf opg2
Setting base /data0/x5-5-0/rls
>
> upgrade.sh

  upgrade.sh  Upgrade from V5.4 to V5.5


  Pass

    savedirectory  Save directory volume.
    classvolumes   Create loadfiles for classvolumes.
    renamedb       Rename old databases.
    loaddb         Load dumpfiles.
    compile        Compile all plcprograms in the database
    createload     Create new loadfiles.
    createboot     Create bootfiles for all nodes in the project.
    createpackage  Create distribution packages for all nodes in the
                   project.

-- Upgrade opg2

Enter start pass [savedirectory] >

------------------------------------------------------------------------
 Pass save directory volume
------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Processing line: 270
-- Building volume directory
I Volume directory loaded
I Database opened /data0/pwrp/pwrtestloc/src/db/directory.wb_load
-- Saving file /data0/pwrp/pwrtestloc/src/db/directory.wb_load ->
/data0/pwrp/pwrtestloc/src/db/directory.wb_load.1
%WNAV-E-MSG, Session saved

------------------------------------------------------------------------
Pass create structfiles and loadfiles for classvolumes
------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
ls: cannot access /data0/pwrp/opg2/src/db/*.wb_load: No such file or
directory

------------------------------------------------------------------------
Pass rename old databases
```

```
------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Saving file /data0/pwrp/opg2/src/db/volopg.db ->
/data0/pwrp/opg2/src/db/volopg.db.1

------------------------------------------------------------------------
Pass load database
------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Loading volume volopg
...
-- Processing line: 57
-- Building volume directory
I Volume directory loaded
I Database opened /data0/pwrp/opg2/src/db/directory.wb_load
-- Processing line: 200
-- Building volume VolOpg
I Volume VolOpg loaded
Berkeley DB 4.6.21: (September 27, 2007)
info put: 0
Berkeley DB 4.6.21: (September 27, 2007)
info get: 0
int rc =  m_txn->abort(): 0

------------------------------------------------------------------------
Pass compile plcprograms
------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
...
Berkeley DB 4.6.21: (September 27, 2007)
info get: 0
I Database opened /data0/pwrp/opg2/src/db/volopg.db
-- Plc window generated            F1-Z1-Plc-W
-- Plc window compiled for x86_linux optimized -O3 F1-Z1-Plc-W
-- Plc plcpgm compiled for x86_linux optimized -O3 F1-Z1-Plc
-- Plc window generated            F1-Z2-Plc-W
-- Plc window compiled for x86_linux optimized -O3 F1-Z2-Plc-W
-- Plc plcpgm compiled for x86_linux optimized -O3 F1-Z2-Plc

------------------------------------------------------------------------
Pass create loadfiles
------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Removing old loadfiles
rm: cannot remove `/data0/pwrp/opg2/bld/common/load/ld_vol*.dat': No
such file or directory
...
Berkeley DB 4.6.21: (September 27, 2007)
info get: 0
I Database opened /data0/pwrp/opg2/src/db/volopg.db
-- Building archive for volume: 000_001_001_012
-- Archive built for volume: 000_001_001_012

-- Working with load file volume 'VolOpg'...
-- Open file...
```

```
-- Successfully created load file for volume 'VolOpg'
-- 26 objects with a total body size of 21976 bytes were written to new
file.

Before this pass you should compile the modules included by ra_plc_user.


------------------------------------------------------------------------
Pass create bootfiles
------------------------------------------------------------------------
Do you want to continue ? [y/n/go] y
-- Creating bootfiles for all nodes

Proview is free software; covered by the GNU General Public License.
You can redistribute it and/or modify it under the terms of this
license.

Proview is distributed in the hope that it will be useful
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

-- Creating bootfile for node opg
      plc_opg_0507_00011
-- Plc thread generated priority 0, scantime   0.10000 s, 2 plcpgm's
-- Plc process compiled for x86_linux optimized -O3 Dummy
-- Plc program linked for x86_linux  node plc_opg_0507
-- Creating bootfile for node aristotle
      plc_aristotle_0517_00011
-- Plc thread generated priority 0, scantime   0.10000 s, 2 plcpgm's
-- Plc process compiled for x86_linux optimized -O3 Dummy
-- Plc program linked for x86_linux  node plc_aristotle_0517

-- The upgrade procedure is now accomplished.

setdb is obsolete
>
>
```