**The Spinsolve-Expert**

**Internal Procedure and Parameter description**

Contents

[Open new editor 1](#_Toc52051517)

[Open new 1D Plot 2](#_Toc52051518)

[Open new 2D Plot 2](#_Toc52051519)

[Open new 3D Plot 3](#_Toc52051520)

[Show/hide temporary windows 3](#_Toc52051521)

[Show next window 4](#_Toc52051522)

[Show last window 4](#_Toc52051523)

[Edit pulse program menus 5](#_Toc52051524)

Spinsolve Expert from version 1.40 onwards uses classes to store information and procedures. Many of these classes are globally accessible which means you can access them from any procedure. This means it is unnecessary to pass parameters as arguments to procedures as was done with earlier versions. One advantage of this is simpler code but more importantly it means that new parameters become available without modification of multiple procedures resulting in fewer potential bugs.

This document consists of a list of the most important class parameters (members) within the SpinsolveExpert application organised by function.

## Globally accessible classes

There are 9 globally accessible class instances in Expert which have useful functionality to the end user

gBatch ... batch processing

gData .... relating to

gExpt .... running experiments

gMenus ... modifying the user defined menus

gParam ... experiment parameter

gPlot .... displaying data

gProc ....

gSample ..

gSeq .....

gView .... generating and controlling the user interface

The classes these are instantiated from are defined in the folder

<prospa\_install\_folder>\Macros\Spinsolve-Expert\Classes

The class file have the same name as the instances except the prefix is ‘se’ (SpinsolveExpert) not ‘g’ (global). For example the seView file instantiates gView.

Classes in Prospa are structures which can also contain procedures. You can access a class member using the syntax

class\_instance->class\_member

class\_member can a variable or a procedure. Examples

pr gView->dirList

This lists all the data directories in the current day of the history list.

gView->g1

This is the first plot object in the user interface. To draw something into this plot (if it is visible) you could type

gView->g1->subplot(1,1)->plot(noise(100))

## SpinsolveExpert startup

The entry point for the program is the Prospa executable file SpinsolveExpertInterface.pex which is stored in the folder

<prospa\_install\_folder>\Macros\Spinsolve-Expert

In this file the above classes are instantiated and cached for fast access. Then the user interface is generated by calling the procedure

gView->makeUI()

Because the instantiated class name is not the same as the file name it is necessary to make a link between the two for the editors to enable them to find the various class procedures when you control-double click on the procedure name. This is done in with the procedure:

SpinsolveExpertInterface.pex:defineClassAssociations

## gView – the View class

This class defines the majority of the user interface and the various callbacks which occur when you interact with the user interface. The user interface is complex so it is broken in several further parts – gPlots handles graphs and images and gBatch handles the batch list.

The useful procedures and parameters here are:

Procedures

# History

self->historyCallback = "seView:historyCallback"

self->loadHistoryList = "seView:loadHistoryList"

self->renameExperiment = "seView:renameExperiment"

self->removeFolderFromHistory = "seView:removeFolderFromHistory"

self->historyMenuCallback = "seView:historyMenuCallback"

# Connection

self->checkUSBConnection = "seView:checkUSBConnection"

# Modify UI

self->disableControls = "seView:disableControls"

self->dragNDropMenu = "seView:dragNDropMenu"

self->enableControls = "seView:enableControls"

self->exitProcedure = "seView:exitProcedure"

self->getUIVersion = "seView:getUIVersion"

self->loadPar = "seView:loadPar"

self->makeSplashScreen = "seView:makeSplashScreen"

self->restoreInterface = "seView:restoreInterface"

self->savePar = "seView:savePar"

self->toggleBatchMode = "seView:toggleBatchMode"

self->toggle1DLegend = "seView:toggle1DLegend"

self->selectInterface = "seView:selectInterface"

# Make UI

self->addNewPPPath = "seView:addNewPPPath"

self->addPPMenus = "seView:addPPMenus"

self->addTestMenus = "seView:addTestMenus"

self->defineBatchPanel = "seView:defineBatchPanel"

self->defineCLI = "seView:defineCLI"

self->defineDividers = "seView:defineDividers"

self->defineExperimentControl = "seView:defineExperimentControl"

self->defineFileControls = "seView:defineFileControls"

self->defineGraph = "seView:defineGraph"

self->defineHistoryPanel = "seView:defineHistoryPanel"

self->defineMainWindow = "seView:defineMainWindow"

self->defineMenus = "seView:defineMenus"

self->defineParameterPanel = "seView:defineParameterPanel"

self->definePostProcessing = "seView:definePostProcessing"

self->defineStatusAndViewPanel = "seView:defineStatusAndViewPanel",

self->makeUI = "seView:makeUI"

# General callbacks

self->changeViewDate = "seView:changeViewDate"

self->copyAllPlots = "seView:copyAllPlots"

self->gotoExptFolder = "seView:gotoExptFolder"

self->openDataFolder = "seView:openDataFolder"

self->openPPEditor = "seView:openPPEditor"

self->makeShortcut = "seView:makeShortcut"

self->processMessages = "seView:processMessages"

self->readAcqPar = "seView:readAcqPar"

self->searchHistory = "seView:searchHistory"

self->showExperimentHelp = "seView:showExperimentHelp"

self->showReleaseNotes = "seView:showReleaseNotes"

self->updateProjectPath = "seView:updateProjectPath"

self->viewHelp = "seView:viewHelp"

self->variableList = "============================================="

# UI dimensions

self->historyWidth = 290 # Some groups of controls rely

self->parameterWidth = 285 # on these parameters to know

self->fileControlHeight = 100 # where they will be placed and they

self->progressHeight = 84 # may not have been generated yet.

# Other

self->historyLoading = 0 # A flag to prevent selecting the history entries too fast

self->nrGraphs = 10 # The number of 1D and 2D plots to make

self->nr1DPlots = 6 # The number of 1D plots to make

self->customerMode = 1 # Disables some functions

self->dirList = list(0) # History list directories

# Window relayout variables

self->userCtrlMaxRng = [1,199] # Maximium user controls range

self->staticMaxRng = [200,499] # Maximium static controls range

self->menuMaxRng = [500,699] # Maximum menus range

self->staticCtrlRange = [-1,-1] # Actual range of static controls (Updated in makeUI)

self->userMenusRange = [-1,-1] # Actual control number range of user-define menus

self->IsMaximised = "normal" # Startup state: maximized or normal

self->displayMode = "normal" # How the UI is organised

self->nrDefaultExptMenus = 7 # Number of default menus after user supplied list

self->batchRange = [-1,-1] # Range of controls in batch group

self->lastBatchDividerPos = "wh\*0.4"

# Initialise member variables

self->cn = 0 # Control number counter

self->wn = 0 # Main window number

# Version

self->versionNr = "1.40.8" # Program version number

(self->coreTitle, self->tester, self->icon) = expertName(self->versionNr) # Title/tester/icon