

DPlot User Manual

© 2014 HydeSoft Computing, LLC

DPlot User Manual

by HydeSoft Computing, LLC

DPlot is a general purpose graphing program designed for scientists, engineers, and students. It features multiple scaling types, including linear, logarithmic, and probability scales, as well as several special purpose graphs: tripartite grids (shock spectra), grain size distribution plots, polar plots, contour plots of 3D data, and box-and-whisker plots and dot graphs of one or more 1D samples. Manipulation functions include FFT, filtering, and smoothing. Data can be input via file, copied from the clipboard, generated by an equation within DPlot, sent to DPlot from Excel via the supplied Add-In, or sent to DPlot via dynamic data exchange. Source code for example Visual Basic and C demo driver programs are included.

DPlot User Manual

© 2014 HydeSoft Computing, LLC

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Printed: March 2014 in Vicksburg, Mississippi.

Production

David Hyde

Chief of Security

Diesel

Table of Contents

Part I	Getting Started	17
1	Getting Started with XY Plots	18
	Getting Started with XY Plots - Data From a Disk File	
	Getting Started with XY Plots - Data From Another Program	
	Getting Started with XY Plots - Data From an Equation	
	Getting Started with XY Plots - Entering Data Values	30
2	Getting Started with XYZ Surface Plots	31
	Getting Started with XYZ Surface Plots - Data From a Disk File	31
	Getting Started with XYZ Surface Plots - Data From Another Program	33
	Getting Started with XYZ Surface Plots - Data From an Equation	
	Getting Started with XYZ Surface Plots - Entering Data Values	
3	Getting Started with XYZ Scatter Plots	36
	Getting Started with XYZ Scatter Plots - Data From a Disk File	
	Getting Started with XYZ Scatter Plots - Data From Another Program	
	Getting Started with XYZ Scatter Plots - Data From an Equation	
	Getting Started with XYZ Scatter Plots - Entering Data Values	
4	Getting Started with 1D Plots	39
	Getting Started with 1D Plots - Data From a Disk File	40
	Getting Started with 1D Plots - Data From Another Program	41
	Getting Started with 1D Plots - Entering Data Values	
Part II	How do I?	44
1	Example plots	45
	EX01.GRF - The basics	
	EX02.GRF - Multiple dependent scales	
	EX03.GRF - Polar plots and number formatting	50
	EX04.GRF - Multiple independent Y scales	
	EX05.GRF - Tripartite plots	
	EX08.GRF - Bar chart	56
	EX09.GRF - Surface plot from an equation	58
	EX10.GRF - Point labels	60
	EX14.GRF - Triangle plot	62
	EX16.GRF - Surface plot with concave boundary	
2	Data Entry	68
	How do I create a plot from scratch?	68
	How do I import data from another Windows application?	69
	How do Limport data from a DOS application?	60

	How do I import data from a DOS application?	69
3	Data Files	69
	How do I open a CSV (comma-separated values) file?	
	How do I save a plot to an EPS (Encapsulated PostScript) file?	
	How do I save a plot to a PDF (Portable Document Format) file?	
	How do I save a collection of files to a different file format?	
	How do I open and/or print a file from Explorer?	
	How do I get rid of the prompt to save changed plots?	

4	Algebra	74
	How do I graph an equation?	74
	How do I make a graph of a circle?	75
	How do I graph an inequality?	
	How do I graph the solution to a system of inequalities?	80
	How do I find the area under a curve?	84
5	Printing	86
	How do I print colors and/or grav shades on a B/W printer?	86
	How do I print several documents unattended?	
	How do I print multiple documents on the same page?	86
	How do I print blank graph paper?	86
6	Appearance	88
	How do I change the direction of the coordinate axes?	88
	How do I change the orientation of the numbers along each axis?	88
	How do I draw a smooth curve through sparse data?	89
	How do I create a series of plots with the same settings (scaling, labels, colors, etc.)?	90
	How do I change the line widths used for the coordinate axes and/or the grid?	91
	How do I change the format of the numbers appearing on the X and Y axes?	91
	How do I force DPlot to create a triangular mesh of my 3D points that is not convex?	91
	How do I create a scatter plot?	92
	How do I replace the numbers along the axes with custom text?	92
_	How do I set error bars with unique error values?	
7	Scaling	
	How do I create a polar plot?	97
	How do I create a bar chart?	98
	How do I create a grain size distribution plot?	102
	How do I specify a tick mark interval?	107
	How do I force scaling on the X axis to be equal to the scaling the Y axis?	
8	Startup Options	108
	How do I specify options on the DPIot command line?	108
9	Miscellaneous	111
	How do I pan from side-to-side (or up-down) on a plot?	111
	How do I temporarily hide one or more curves on a crowded document?	111
	How do I distribute my DPlot files to others?	112
	How do I get help within DPlot?	112
Part III	License and Ordering Info	114
1	Evaluation License	114
2	Copyright, License, & Warranty	115
3	DPlot Ordering Information	116
4	DPlot Order Form	118
5	Register DPlot	120
Part IV	Menu Commands	122
. uit IV		
1		
	New command	123
	Open command	
	DFIUL FIIH Bandix Format Files	

	Pacific Data Recorder Old Format	135
	Pacific Data Recorder New Format	136
	Nicolet Waveform File Specification Format	138
	HDAS File Format	
	File Import Plugin Modules	
	Append command	
	Save command	
	Save As command	
	Image Properties	146
	Close command	
	Get Preferences command	
	Save Preferences command	
	Print command	
	Macro command	
	Creating and/or running a macro	
	Resize Arrays command	
	1. 2. 3. 4 command	
	Exit command	153
2	Edit Menu	153
~		100
	Copy Picture command	157
	Copy Data Values command	158
	Copy Dimensions command	158
	Paste command	159
	Paste Special command	
	Insert Link command	
	Edit Data command	
	Decimate command	
	Filter data command	
	Low Pass Filters	166
	Baseline Shift command	
	List Editions command	167
	Smooth command	167
	Equal Intervals command	
	Operate on X command	171
	Operate on Y command	
	Operate on Z command	172
	Edit Mode command	
	Remove Random Spikes command	
	Swap X, Y command	
3	Text Menu	
		475
	Langendy shale command	
	Legend/Labels command	
	Label Points command	
	Fonts command	
	Add/Edit Note command	
	lext Placeholders	
	X,Y,Z Labels	
4	Options Menu	190
	Linear/Log Scaling	
	Triangle Plot Options	
	Polar Plot Options	
	Bar Chart Options	
	Grain Size Options	

	Tripartite Options	210
	Grid Lines or Box command	211
	Symbol/Line Styles command	212
	Extents/Intervals /Size command	214
	Square log cycles command	217
	Log scale tick labels command	217
	Multiple Axes command	217
	Fill Between Curves command	220
	Number Format command	224
	Colors command	227
	Amplitude Limits command	228
	Data Processing command	231
	Bubble Plot command	232
	Vector Plot command	233
	OHLC Chart command	235
	Error Bars command	240
	Reference Lines command	241
	Generate Mesh command	243
	Contour Options command	244
	Scatter Plot Options	253
	Symbol Styles command (3D scatter plots)	255
	Box and Whisker Plot Format	256
	Box and Whisker Plot Attributes	
	Dot Graph Format	257
	Define Boundary command	258
	Background Image command	260
		262
	Insert Image command	
	Insert Image command General command	
5	Insert Image command General command Generate Menu	
5	Insert Image command	
5	Insert Image command General command Generate Menu Average All Curves command	262 263 267 268 268
5	Insert Image command General command Generate Menu Average All Curves command Median of All Curves command	262 263 267 268 268 268
5	Insert Image command General command Generate Menu. Average All Curves command Median of All Curves command Polynomial Curve Fit command	262 263 267 267 268 268 268 268
5	Insert Image command General command Generate Menu Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve	262 263 267 268 268 268 268 268 272
5	Insert Image command General command Generate Menu Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve	262 263 267 268 268 268 268 268 268 272 272
5	Insert Image command General command Generate Menu Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command	262 263 267 268 268 268 268 268 268 272 273 273 273
5	Insert Image command General command Generate Menu. Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command	262 263 267 268 268 268 268 268 268 272 273 274 273
5	Insert Image command General command Generate Menu. Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command	262 263 267 268 268 268 268 268 268 272 273 274 275 278
5	Insert Image command General command Generate Menu. Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command X=f(T), Y=g(T) command	262 263 267 268 268 268 268 268 272 273 274 273 274 275 278 278 279
5	Insert Image command General command Generate Menu. Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command X=f(T), Y=g(T) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command	262 263 267 268 268 268 268 268 272 273 273 274 275 278 279 279 279
5	Insert Image command General command Generate Menu. Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command X=f(T), Y=g(T) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command	262 263 267 268 268 268 268 268 272 273 273 274 273 274 275 278 279 281 282
5	Insert Image command General command Generate Menu. Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command X=f(X,Y1,Y2,) command	262 263 267 268 268 268 268 268 272 273 274 273 274 275 278 279 281 282
5	Insert Image command General command Generate Menu Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command Z=f(X,Y) command X=f(T), Y=g(T) command X=f(T), Y=g(T), Z=h(T) command Y=Poly(Y) command Y=f(X,Y1,Y2,) command Histogram command	262 263 267 268 268 268 268 268 272 273 274 273 274 275 278 279 281 282 282 282
5	Insert Image command General command Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command X=f(T), Y=g(T) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command Y=Poly(Y) command Y=f(X,Y1,Y2,) command Histogram command Average Curve with Error Bars	262 263 267 268 268 268 268 268 272 273 274 275 274 275 278 279 281 282 282 282
5	Insert Image command General command Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command X=f(T), Y=g(T) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command Y=Poly(Y) command Y=f(X,Y1,Y2,) command Histogram command Average Curve with Error Bars XY Plot from 3D Surface command	262 263 267 268 268 268 268 272 273 274 275 274 275 278 279 281 282 282 282 282
5	Insert Image command General command Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command Y=f(X,Y1,Y2,) command Y=f(X,Y1,Y2,) command Histogram command Average Curve with Error Bars XY Plot from 3D Surface command	262 263 267 268 268 268 268 268 272 273 273 274 273 274 275 278 279 281 282 282 282 282 282 282
5	Insert Image command General command Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command Y=f(X) command X=f(T), Y=g(T) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command Y=f(X,Y1,Y2,) command Histogram command Average Curve with Error Bars XY Plot from 3D Surface command Find Best-Fit Plane command	262 263 267 268 268 268 268 268 272 273 274 273 274 275 278 279 281 282 282 282 282 282 282 284 282
5	Insert Image command General command Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command X=f(X) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command Y=Poly(Y) command Y=f(X,Y1,Y2,) command Y=f(X,Y1,Y2,) command Histogram command Average Curve with Error Bars XY Plot from 3D Surface command Find Best-Fit Plane command	262 263 267 268 268 268 268 268 272 273 274 274 275 278 279 281 282 282 282 282 282 284 287 289 290 291
5	Insert Image command General command Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command X=f(T), Y=g(T) command X=f(T), Y=g(T) command X=f(T), Y=g(T), Z=h(T) command X=f(T), Y=g(T), Z=h(T) command Y=Poly(Y) command Y=Poly(Y) command Histogram command Average Curve with Error Bars XY Plot from 3D Surface command Find Volume Under Surface command Subtract a Plane command Z=f(Z,Z2) command	262 263 267 268 268 268 268 268 272 273 274 274 275 278 279 281 282 282 282 282 282 284 289 290 291 291
5	Insert Image command General command Generate Menu. Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve FFT command Y=f(X) command X=f(T), Y=g(T) command X=f(T), Y=g(T), Z=h(T) command Y=Poly(Y) command Y=Poly(Y) command Y=Poly(Y) command Y=f(X,Y1,Y2,) command Histogram command Average Curve with Error Bars XY Plot from 3D Surface command Find Volume Under Surface command Find Best-Fit Plane command Subtract a Plane command Z3=f(Z1,Z2) command	262 263 267 268 268 268 268 272 273 274 275 274 275 278 279 281 282 282 282 282 282 284 289 290 291 291 291
5	Insert Image command General command Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve FFT command Y=f(X) command X=f(T), Y=g(T) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command Y=Poly(Y) command Y=Poly(Y) command Y=f(X,Y1,Y2,) command Histogram command Average Curve with Error Bars XY Plot from 3D Surface command Find Volume Under Surface command Subtract a Plane command Z=f(Z,Z2) command Average Surfaces command Switch Independent Variable command	262 263 267 268 268 268 268 268 272 273 274 275 278 279 281 282 282 282 282 284 287 289 290 291 291 291 291
5	Insert Image command General command Generate Menu. Average All Curves command Median of All Curves command Polynomial Curve Fit command Integrate a Curve Differentiate a Curve FFT command X=f(T), Y=g(T) command Z=f(X,Y) command X=f(T), Y=g(T), Z=h(T) command X=f(T), Y=g(T), Z=h(T) command Y=Poly(Y) command Y=Poly(Y) command Y=f(X,Y1,Y2,) command Histogram command Average Curve with Error Bars XY Plot from 3D Surface command Find Volume Under Surface command Find Volume Under Surface command Subtract a Plane command X=f(Z,Z), Z=h(Z),	262 263 267 268 268 268 268 272 273 274 275 278 279 281 282 282 282 282 284 289 290 291 291 291 291 291
5	Insert Image command General command Generate Menu	262 263 267 268 268 268 268 268 272 273 274 274 275 278 279 281 282 282 282 282 284 287 289 290 291 291 291 291 291 293 293 294 294
5	Insert Image command General command Generate Menu	262 263 267 268 268 268 268 268 272 273 274 274 275 278 279 281 282 282 282 282 282 282 282 282 283 290 291 291 291 291 291 293 293 293 293

	Scan X, Y command	298
	List Peaks command	299
7	View Menu	
	Playback command	302
	Toolbar description	303
8	Window Menu	
Dent V		206
Part v	Programmer's Reference	300
1	Using WinExec to plot an existing file	
2	Sending data to DPlot from another application	
	AAxisLabel macro command	
	About command	315
	Activate command	315
	AddCurves command	316
	AppHide command	316
	AppMaximize command	316
	AppMinimize command	316
	AppRestore command	316
	AppShow command	316
	AutoScale command	316
	AutoSize command	
	AverageAll command	
	AverageAndReplace command	
	Axes_At_Zero command	
	AxisArrowheads command	
	Axistolor command	
	Axisi ype command	
	BarChartStyle command	
	Barwidur command	
	Baseinie Sinit Command	
	BkBlotColor command	319
	Bubbles command	
		320
	Cantion command	320
	ClinY command	
	ClearEditElag command	321
	Color command	
	ColorBasis command	
	ColorCvcle command	
	ColumnsAre command	
	Combine Curves command	
	Compare command	
	Contour3D command	
	ContourAxes command	
	ContourBarWidth command	
	ContourColorScheme command	
	ContourCustomColors command	
	ContourCustomLevels command	
	ContourDrawSides command	
	ContourGrid command	
	ContourLabelLines command	

ContourLegend command	. 327
ContourLegendFrequency command	. 327
ContourLevels command	. 327
ContourLighting command	. 328
ContourMethod command	. 328
ContourResetExtents command	. 331
ContourScales command	. 331
ContourView command	. 332
ContourViewChange macro command	. 333
CopyPicture command	. 334
CopyText command	. 334
CurveFit command	. 334
CurveLabel command	. 334
DataProcess command	. 334
DataProcessLabels command	. 335
DateFormat command	. 335
DecimalDigits command	. 335
DeleteLabel command	. 335
DeferMinMaxCheck command	. 336
DeleteImage command	. 336
DeleteNote command	. 336
DeleteRefLine command	. 336
DeleteThinTriangles command	. 337
Dense GridLines command	. 337
Differentiate command	. 337
Directory command	. 337
DivideCurves command	. 338
DocHide command	. 338
DocMaximize command	. 338
DocMinimize command	338
DocRestore command	. 339
DocShow command	. 339
EditAddToX command	339
EditAddToX command	339
EditDecimate command	339
EditEqualInts command	339
EditErase command	330
EditErase Data command	3/0
EditEllar command	340
EditMultinlyY command	2/1
EditMultinlyV command	3/1
EditNormalize command	3/1
	2/1
	2/1
EditOperate 7 command	2/1
EditPamaya Spikas command	241
EditPamayaTrand command	242
	. 34Z
EditSmooth command	2/2
Euromour command	. 343 242
	. 343 240
	. 343
Eulowaph r command	. 343
cut truncate command	. 343
Enable command	. 344

ErrorBar command	. 344
ErrorBarPenWidth command	. 344
ErrorBarWidth command	. 344
ErrorMessages command	. 345
FFT command	. 345
FFTMethod command	. 345
FFTPhase command	. 345
FileAppend command	. 345
FileArrays command	. 346
FileClose command	. 346
FileExit command	. 347
FileMacro command	. 347
FileNew command	. 347
FileNewEx command	. 347
FileOpen command	. 347
FilePageSetup command	. 349
FilePrint command	. 350
FilePrintMultiple command	. 351
FileSaveAs command	. 352
FileType command	. 354
FillBetween command	. 355
FillStyle command	. 356
FindIntersections command	. 356
FontPoints command	. 356
ForAllDocuments command	. 356
ForFilesIn command	. 357
FunctionOfXY1Y2 command	. 358
FXY1Y2_XFrom command	. 358
GeneralOptions command	. 358
GenerateMesh command	. 359
GetPreferences command	. 360
GrainSizeFlags command	. 360
GrainSizeFlagsEx command	. 362
GrainSizeOther command	. 362
GridLines command	. 362
GridLinesOnTop command	. 363
GridLineType command	. 363
HelpContents command	. 363
HelpSearch command	. 363
HideCurve command	. 363
Histogram command	. 364
IndependentTicks command	. 364
InfoInterpolate command	. 364
InfoScan command	. 365
Insertimage command	. 365
InsertLink command	. 365
InsertXY command	. 365
Integrate command	. 366
InvertAxis command	. 366
LabelDirection command	. 366
Legend command	. 366
LegendParams command	. 366
LegendShow command	. 367
LineType command	. 367

LineWidth command	368
ListPeaksOptions command	368
LogTicks command	368
Lump command	368
ManualScale command	369
Medians command	369
MenuHide command	369
MenuShow command	370
MenuUpdates command	370
MoveCurves command	370
MultipleXAxes command	370
MultipleYAxes command	371
MultiplyCurves command	372
NextDocument command	372
NextFile command	372
NoteButtonText command	372
Note Options command	373
NotePosition command	373
Note Tabstops command	373
Note Text command	373
NSEW command	373
NumberFormat command	374
NumTicks command	375
OHLC macro command	375
OverlayDocument command	375
Paste command	376
PasteLink command	376
Pause command	376
PolarPlotStyle command	376
PostponeRedraw command	377
Preserve3DScaleFactor command	377
ProbabilityFlags command	378
RefLine command	378
RemoveLink command	379
ReorderCurves command	379
Repaint command	379
Request function	380
RunPlugin command	380
SavePreferences command	380
SaveUndo command	381
Scale command	381
ScatterPlotLines command	382
SelectCurve command	383
SetBackgroundImage command	384
SetBitmapDims command	385
SetClientRect command	385
SetDecimalSymbol command	385
SetFilename macro command	386
SetImageCrop command	386
SetMetafileDims command	386
SetOutputWindow command	386
SetPanFraction command	388
SetPluginImageDims command	388
SetWindowPos command	390

Shell command	. 392
ShowCaption command	. 392
ShowCurve command	. 393
ShowDate command	. 393
Size command	. 393
SplitDateTime command	. 394
SquareLogCycles command	. 394
StartStop command	. 394
Stat_AddData command	. 395
Stat_AddGroup command	. 395
Stat_Box command	. 395
Stat_BWOptions command	. 395
Stat_DotOptions command	. 396
Stat_DrawLegend command	. 396
Stat_GrandMean command	. 396
Stat_GroupInfo command	. 397
Stat_LimitsAsCL command	. 397
Stat_MeanFormat command	. 397
Stat_Outliers command	. 397
Stat_PlotType command	. 398
Stat_RefLine command	. 398
Stat_Rotated command	. 398
Stat_Scaling command	. 398
Stat_Whiskers command	. 398
SubtractCurves command	. 399
SwitchX command	. 399
SymbolClip command	. 401
SymbolCycleColors command	. 402
SymbolFillBackground command	. 402
SymbolFrequency command	. 402
SymbolOpacity command	. 402
SymbolOutline command	. 404
SymbolOutlineColor command	. 404
SymbolSize command	. 405
SymbolSizeUnits command	. 405
SymbolType command	. 405
TextFont command	. 407
TextNote command	. 407
TextNote3D command	. 408
TextNoteEx command	. 409
TextNoteLeader command	. 411
TextPointLabel command	. 411
TextPointLabel3D command	. 412
TextPointLabelLeader command	. 413
TickInterval command	. 413
TickMarks command	. 414
TimeFormat command	. 414
Title1 command	. 414
Title2 command	. 414
Title3 command	. 414
Title4 command	. 415
TriangleFlags command	. 415
TripartiteFlags command	. 415
UseNameAsLegend command	. 415

	VectorPlot command	
	VectorPlotColors command	416
	VectorPlotOptions command	
	ViewRedraw command	
	ViewStatusBar command	417
	ViewToolBar command	
	WindowCascade command	
	WindowTile command	
	XAxisLabel command	
	X2AxisLabel command	
	XLabel command	
	XY command	
	XYLabel command	419
	XYXY command	
	XYZ command	
	XYZ_To_XY command	
	XYZEx command	
	XYZRegen command	
	Y2AxisLabel command	
	Y2TopDown command	
	YAxisHorizontal command	
	YAxisLabel command	
	YCutoff command	
	ZAxisLabel command	
3	Plugins for File Import, Export and Data Manipulation	
-	······································	
Dart VI	Poforonco	430
Fait VI		-50
Fait VI	Keyboard Shortcuts	
1 2	Keyboard Shortcuts Right-Click and Mouse Wheel Operations	
1 2 3	Keyboard Shortcuts Right-Click and Mouse Wheel Operations Functions	430
1 2 3	Keyboard Shortcuts Right-Click and Mouse Wheel Operations Functions	430
1 2 3	Keyboard Shortcuts Right-Click and Mouse Wheel Operations Functions ABS function	430
1 2 3	Keyboard Shortcuts Right-Click and Mouse Wheel Operations Functions ABS function ACOS function	+30
1 2 3	Keyboard Shortcuts Right-Click and Mouse Wheel Operations Functions ABS function ACOS function ACOSH function	+30
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function	+30
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASINH function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASINH function ATAN function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASIN function ATAN function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASINH function ATANY function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function ASIN function ASIN function ATAN function ATANH function BESSEL I function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function ASIN function ASIN function ATAN function ATAN1 function BESSELJ function BESSEL function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASIN function ATAN function ATAN2 function BESSELJ function BESSELY function CDE function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASIN function ATAN function ATAN2 function BESSELJ function BESSELY function CDF function CELL function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASIN function ATAN function ATAN2 function BESSELJ function BESSELY function CDF function COS function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASIN function ATAN function BESSELJ function BESSELJ function CDF function CDF function COS function COS function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASIN function ATAN function ATANY function BESSELJ function BESSELJ function CDF function COS function COS function DATE function	430
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASIN function ATAN function ATAN function BESSELJ function BESSELJ function CDF function CDF function COS function COS function DATE function	+30
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASIN function ATAN function ATAN function BESSELJ function BESSELJ function CDF function COS function COS function BESSELY function COS function DATE function DATE function ERF function	+30
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ACOSH function AND function ASIN function ASIN function ATAN function ATAN function BESSELJ function BESSELJ function CDF function COS function COS function BESSELY function COS function DATE function DATE function ERF function ERF function	430 432 439 441 441 441 442 443 443 444 445 445 445 445 445 445 445
1 2 3	Keyboard Shortcuts. Right-Click and Mouse Wheel Operations. Functions. ABS function ACOS function ASIN function ASIN function ATAN function ATAN2 function BESSELJ function BESSELJ function CDF function COS function COS function DATE function DATE function ERF function ERF function ERF function ERF function	430 432 432 439 441 441 441 442 443 443 443 444 445 445 445 445 445 445

	FPOWER function	453
	GCD function	453
	LCM function	454
	LOG function	454
	LOG10 function	454
	MAX function	454
	MIN function	454
	MOD function	455
	MONTH function	455
	MROUND function	455
	NORMDIST function	456
	NORMDISTCDF function	456
	NORMINV function	457
	NORMRAND function	457
	NOT function	459
	OR function	460
	PMT function	461
	RANDBETWEEN function	463
	RND function	
	ROUND function	
	SIGN function	
	SIN function	
	SINH function	
	SQRT function	
	SUM function	
	TAN function	
	TANH function	468
	WEEKDAY function	469
	WEEKNUM function	
	XOR function	470
	YEAR function	
4	Error Messages	471
5	Known Problems	472
6	Acknowledgements	
Part VII	Technical Support	476
1	Contact information	476
2	Notes for DPlot Jr and DPlot Viewer users	477
		400
	I ransferring data	400
1	Using the DPlot Interface Add-In for Microsoft Excel	480
	XYXY Add-In command	
	XYYY Add-In command	
	X,Y,Label command	
	Bar chart command	
	Bar chart, data in rows command	490
	OneCurvePerRow Add-In command	491
	XYZSurface Add-In command	
	XYZScatter Add-In command	493
	ZGrid Add-In command	495
	OneD Add-In command	495

	Contents	15
2 Transferring DPlot plots to other applications		496
3 Transferring DPlot data to other applications		497
Index		501

Г



1 Getting Started

Here you will find help for first-time DPlot users. You might also want to check the <u>How Do I?</u> section for answers to more specific questions.

What type of plot do you want to create?





	100			
	-45			
	-110			
	-05			<u> </u>
		· ·		H -
	-001	6 E 88 +		+•
I	-612	• • • •	H	<u> </u>
	-49		Н	
	-00		-	- +
	-80		H	
	111	• •• HE		
	111	-	H	<u> </u>

1.1 Getting Started with XY Plots



What is the source of your data?

A file on disk

Excel or another program

An equation

No external source - enter data in DPlot

1.1.1 Getting Started with XY Plots - Data From a Disk File

Select the *Open* command on the *File* menu. DPlot currently has direct support for 10 different file formats used for XY plots (types A-J below), plus additional formats supported by any <u>File Import</u> plugins that you might have installed.

Select File Type		×
File <u>Types</u>		
A DPlot File	🔘 <u>G</u> Pacific digitizer, old format	
○ <u>B</u> n (number of points)	○ <u>Н</u> Pacific digitizer, new format	
x(1), y(1), , x(n), y(n)	🖸 📋 Nicolet time domain	
v n dx (assumes x(1)=0) y(1),, y(n)	C J Hardened Data Acquisition System (HDAS)	
D Multiple columns (2D)	○ K 3D - X,YZ columns	
C E Unformatted binary	🔿 L Plugins	
n (number of points) x(1), y(1), , x(n), y(n)	WAV Audio	•
E Pacific digitizer model 9820, Bendix/DNA format	C M 1D Statistics	
- Dondar Drive Tomat		
ОК	Cancel <u>H</u> elp	

Click the Help button on the Select File Type dialog for a full description of each supported file format. DPlot Files are of course files saved by DPlot. Types F-J provide support for files produced by a variety of digital recorders. Chances are if you can use those file formats, you're taking a look at DPlot specifically because of support for those files. By far the most commonly used format when importing data for the first time is type *D Multiple columns (2D)*. This file format includes full support for comma-separated values (CSV) files produced by Microsoft Excel. A full description of DPlot's support for multiple column files, along with several examples, is repeated here (from the Help topic for the <u>Open</u> command): Multiple column text files (type D on the **Select File Type** dialog) may have columns delimited by commas (CSV files, for example), spaces, tabs, or semicolons. If the <u>decimal symbol</u> is set to a comma, columns should be separated by tabs or semicolons. Files may contain up to 30 header lines preceding the data columns (maximum number of header lines can be increased with the <u>General</u> command on the Options menu). DPlot looks for up to 20 successive lines that have the same number of data columns and the same data type in each column, and, if found, decides that this must be where the data starts. Data values may be separated by one or more spaces, a comma, or a tab. DPlot will read up to a maximum of 100 columns of data, restricted to 8192 characters per line.

Column Headings. If the data starts after line 1, DPlot attempts to get column headings from the previous line. Column headings are used as the legend for multiple curves, or the X and Y axis labels for a single curve. For this feature to work as expected with headings containing spaces, the columns must either be comma-separated, tab-separated, or delineated with double quotation marks. Otherwise DPlot will assume the labels are delineated with spaces.

Title Lines. Up to the first 3 lines in the file are used as title lines for the plot, unless these lines consist of numbers and/or spaces only.

Column Interpretation. If the file consists of a single column of data DPlot interprets the data as Y values, setting the corresponding X values to the index of the Y value in the file, starting from 0. By default, if the file consists of more than one column of data, DPlot uses the first column for the X array and subsequent columns as separate Y arrays. For multiple-column files, and/or for files consisting of alternating X,Y columns, you can change this default behavior by checking the **Pick Columns to Plot** box (see below) on the Open dialog box.

Example:



Example (column headings):



Example (column headings, comma-separated):

This is the title line

10

This is the title line X,X²,X³,X⁴ 1,1,1,1 2,4,8,16 3,9,27,81 4,16,64,256 5,25,125,625

 $1.0 \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \end{bmatrix}$ In addition to numbers DPlot will also accept columns consisting of dates, times, date-time pairs, currencies, and percentages. Date format is flexible but entries must be separated by a dash (-) or forward slash (/). If the month is specified as a number, then DPlot assumes the order is m/d/y (or d/m/y if Assume input dates are of the form d/m/y under the <u>General</u> command on the Options menu is checked) unless you use 4 digit years or if the assumed month entry is greater than 12. These date forms (for Jun 8, 2005) are acceptable:

X^2 X^3 X^4

6/8/05 06/08/2005 6-8-2005 2005/6/8 8-Jun-05 June-08-2005

but 'June 8, 2005' is not.

2-digit years less than 90 are interpreted as 21st century dates; 2-digit years greater than or equal to 90 are interpreted as 20th century dates. Of course, to avoid any ambiguity a 4-digit year is preferable.

Time values should be in the form h:m:s AM/PM, using ':' as the separator. Leading zeroes are

acceptable. If the AM or PM designation is omitted, DPlot assumes a 24 hour clock.



Date-time pairs should be in the form 'm/d/yy h:m:s AM/PM'.

Only dollar signs and British pound signs are currently accepted as currency symbols. The currency symbol should precede the value, as in '\$56.23'. If monetary values include a comma for the thousands separator, the entry **must** be surrounded by double quotation marks. This is the same scheme used by Excel when saving CSV files.

Percentages should be followed by a percent sign (%). DPlot divides the number by 100 and uses Percent number formatting on the associated axis when appropriate.

Any other data type will be ignored, but in general will be allowed. If present, blank entries are ignored. (For this feature to work as expected, columns **must** be tab-separated or comma-separated). Entries must not contain commas or tabs unless they are surrounded by double quotation marks. (Microsoft Excel generally surrounds values containing commas with double quotation marks when saving to a CSV file.)

Finally, the data rows may be preceded by an ID character string starting in column 1 that distinguishes this line as data. To make use of this feature check the "Data rows have ID string" box on the Open dialog, and enter the text (up to 8 characters) in the corresponding box. This string is preserved from one session to the next, and once you enter the appropriate string you will be able to drag-and-drop these files onto DPlot without the need to check this box each time.

Example (data row ID string):



If the title line(s) or column headings contain commas, the text to be used should start and end with a double quotation mark (").

Files saved by Microsoft Excel as "comma-separated values" can be read into DPlot using this option.

Campbell Scientific datalogger files

For the most part, Campbell Scientific datalogger files are handled identically to CSV and other multiple-column text files, with a couple of exceptions. DPlot determines that a multiple-column text file is a Campbell datalogger file if:

1) The first seven characters in the file are "TOA5", (including the quotes), and

2) The data is determined to start in line 5, and

3) Lines 2 through 4 contain two times as many quotation marks as there are data columns.

If all of the above tests are met, the file is considered a Campbell Scientific datalogger file. The only significant differences in file handling are:

1) The first title line is taken from the second quoted string in the first line (i.e. "T0A5" is ignored). The second title line is taken from the third quoted string in the first line.

2) Legend entries are taken from the second and third lines in the file. If the entries in the third line are not blank, a comma separator is added between the entry from the second line and the entry from the third line.

3) If the column heading for the second column in the second file line is "RECORD", this column is skipped. Normally the 0-based record number is superfluous. If you **want** the record number to be plotted, you should check the "Pick columns to plot" box.

Pick Columns to Plot

Allows you to specify which columns within a multiple-column file to plot. Up to 20 rows from the selected file are displayed in the read-only box on the left side of the dialog box. Each column is preceded by a heading with the column number as interpreted by DPlot. If the column numbers do not match up with what you expected, or if the displayed text does not start with the first data line, then DPlot had difficulty in determining either the number of columns or the start of the data, or both. For more information see the description of file type D in the <u>Open Command</u> Help topic.

By default, DPlot uses the first column of data in multi-column files for the X axis values. If there is only one column of data, DPlot starts X at 0 and increments by 1 for each additional row. Subsequent columns (or the first column in single-column files) are used for the Y values, each sharing the same X.

To change the column number used for the X axis values, enter a number in the Use column _____ for X Axis box. This entry cannot be a text column. If this number is 0, then DPlot starts X at 0 and increments X by 1 for each additional row of data values.

Select columns to use for the Y values by checking the appropriate check boxes under *Use these columns for* Y. Note that you cannot select a column to serve as both the X Axis values and the Y values for a curve, nor can you select text columns (anything other than numbers, currencies, dates, times, or date-time pairs).

Alternatively, if the file contains alternating X,Y columns, as in:

X(1,1)	Y(1,1)	X(1,2)	Y(1,2)
X(2,1)	Y(2,1)	X(2,2)	Y(2,2)
X(3,1)	Y(3,1)	X(3,2)	Y(3,2)

etc., then you should check the box labelled *Alternating X,Y columns*. In this case DPlot will disable the even-numbered (Y) columns. Checking an X column will automatically cause the corresponding Y column to be checked. *Alternating X,Y columns* cannot be selected if the total number of columns in the file is an odd number, **or** if the file contains any non-numeric columns.

Labels

For X,Y data files containing 3 (and only 3) columns or for 3D data files containing 4 columns, you may specify that the last column contains point labels. For example this data:

0.0, 0.00000000, "X=\$X, Y=\$Y" 0.1, 0.309016994, 0.2, 0.587785252, 0.3, 0.809016994, 0.4, 0.951056516, 0.5, 1.00000000, "X=\$X, Y=\$Y" 0.6, 0.951056516, 0.7, 0.809016994, 0.8, 0.587785252, 0.9, 0.309016994, 1.0, 0.00000000, "X=\$X, Y=\$Y" 1.1,-0.309016994, 1.2,-0.587785252, 1.3,-0.809016994, 1.4,-0.951056516, 1.5,-1.00000000, "X=\$X, Y=\$Y" 1.6,-0.951056516, 1.7,-0.809016994, 1.8,-0.587785252, 1.9,-0.309016994, 2.0, 0.00000000, "X=\$X, Y=\$Y"

will produce this plot:



Scaling

The default scaling for any new XY plot is linear X, linear Y. You can change the scaling (logarithmic X and/or Y, probability scale, polar chart, bar chart, ternary plot, etc.) by right-clicking within the box surrounding the plot and selecting the scaling option from the drop-down menu. Scaling options are also accessible via the Linear/Log Scaling command on the *Options* menu.

Symbols/Line Types

To change the symbol type or line type used by a curve, right-click on the curve and select the appropriate symbol and/or line type. Alternatively, select the <u>Symbol/Line Styles</u> command on the *Options* menu.

Scale Extents, Tick Marks, Plot Size

To change the extents of the X or Y axis, change the tick mark interval used, or to change the physical size of the plot, select Options>Extents/Intervals /Size.

Number Format

To change the number format used on any axis (to dates, date and time groups, multiples of PI, powers of 10, etc.), right-click on any of those numbers and select the appropriate format. To specify that a 1000 separator should be used, to specify the number of decimal places used, or to change the orientation of those numbers, right-click on any of those numbers and select "More number format options...". Alternatively select the <u>Number Formats</u> command on the *Options* menu.

1.1.2 Getting Started with XY Plots - Data From Another Program

Four methods are available for importing data into XY plots from other Windows applications.

The DPlot Interface Add-In for Microsoft Excel

This feature is of course only applicable if the source of your data is a Microsoft Excel worksheet. In general this method is superior to those described below. For more detailed information on Add-In

commands see the Help command on the DPlot menu in Excel. For best results study the example spreadsheet <My Documents>\DPlot\examples.xls.



■ The XYXY command works with alternating X,Y columns



■ The XYYY command is similar to the XYXY command, but X is taken from the first selected column and all subsequently-selected columns are Y values for separate curves.



■ The X,Y,Label command creates one curve on an XY plot with X taken from the first selected column, Y taken from the second, and optional text labels in the third.

The Bar chart command creates a bar chart with non-numeric labels on the X axis and one or more groups of amplitudes.



The Bar chart, data in rows commands is identical, but works with data sets in the same row rather than the same column.



The **Paste** command (Edit menu)

The Paste command is the method most likely to be supported by another Windows application. Once data is pasted from another application, DPlot owns the data. Changes to the data in the source application are not reflected in the DPlot document.

The Paste Special Link option (Edit menu)

The Link option establishes a link with the source of the data. Changes to the data made in the source application are reflected in the linked document. Unlike the Insert Link command, this method requires switching to the source application and back to DPlot.

The Insert Link command (Edit menu)

The Insert Link command performs roughly the same function as the Link option, except that it does not require switching to the source application to initiate the action.

Scaling

The default scaling for any new XY plot is linear X, linear Y. You can change the scaling (logarithmic X and/or Y, probability scale, polar chart, bar chart, ternary plot, etc.) by right-clicking within the box surrounding the plot and selecting the scaling option from the drop-down menu. Scaling options are also accessible via the Linear/Log Scaling command on the *Options* menu.

Symbols/Line Types

To change the symbol type or line type used by a curve, right-click on the curve and select the appropriate symbol and/or line type. Alternatively, select the <u>Symbol/Line Styles</u> command on the *Options* menu.

Scale Extents, Tick Marks, Plot Size

To change the extents of the X or Y axis, change the tick mark interval used, or to change the physical size of the plot, select Options>Extents/Intervals /Size.

Number Format

To change the number format used on any axis (to dates, date and time groups, multiples of PI, powers of 10, etc.), right-click on any of those numbers and select the appropriate format. To specify that a 1000 separator should be used, to specify the number of decimal places used, or to change the orientation of those numbers, right-click on any of those numbers and select "More number format options...". Alternatively select the <u>Number Formats</u> command on the *Options* menu.

1.1.3 Getting Started with XY Plots - Data From an Equation

Two commands on the *Generate* menu are available for generating X,Y values from an equation.

<u>Y=f(X)</u>

Generates a curve whose amplitudes are a function of X.

<u>X=f(T), Y=g(T)</u>

Generates a curve from parametric equations with X and Y both functions of an independent parameter T. This is particularly useful for generating curves that have multiple Y values for the same X - a circle, for example.

All of the function-parsing routines in DPlot have the same basic syntax, capabilities and limitations:

Available two-side-operators are +, -, *, /, ^. Available one-side operators: sin, cos, tan, asin, acos, atan, abs(absolute value), sqrt(square root), log, logh(or log10), exp (inverse natural logarithm), cosh, sinh, tanh, rnd(random integer; argument is highest number that can be the output), fact(factorial), erf (error function), and cdf(cumulative distribution function). A complete list of available functions is shown here. These all take an argument that must be in parentheses.

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute "PI", "pi", or "Pi" in the function with "4*atan(1.0)"

NOTE: Be specific. DPlot does not interpret "2X" as "2*X".

Prevent divide by zero errors

If this box is **unchecked**, DPlot will trap divide by zero and many other math errors (e.g. logarithm of a non-positive value), report the value of x where the error occurred, and will not generate a curve. If this box is **checked**, when DPlot traps a math error it will shift x by a very small number and attempt the calculation again. This feature might be useful for generating curves of certain functions that would normally fail at specific values of x, e.g. sin(x)/x at x=0.

Scaling

The default scaling for any new XY plot is linear X, linear Y. You can change the scaling (logarithmic X and/or Y, probability scale, polar chart, bar chart, ternary plot, etc.) by right-clicking within the box surrounding the plot and selecting the scaling option from the drop-down menu. Scaling options are also accessible via the Linear/Log Scaling command on the *Options* menu.

Symbols/Line Types

To change the symbol type or line type used by a curve, right-click on the curve and select the appropriate symbol and/or line type. Alternatively, select the <u>Symbol/Line Styles</u> command on the *Options* menu.

Scale Extents, Tick Marks, Plot Size

To change the extents of the X or Y axis, change the tick mark interval used, or to change the physical size of the plot, select Options>Extents/Intervals/Size.

Number Format

To change the number format used on any axis (to dates, date and time groups, multiples of PI, powers of 10, etc.), right-click on any of those numbers and select the appropriate format. To specify that a 1000 separator should be used, to specify the number of decimal places used, or to change the orientation of those numbers, right-click on any of those numbers and select "More number format options...". Alternatively select the <u>Number Formats</u> command on the *Options* menu.

1.1.4 Getting Started with XY Plots - Entering Data Values

Select the *Edit data* command on the *Edit* menu to enter new data values for a new curve or edit and/or add data values to an existing curve.

Shortcut:

```
Click is on the toolbar.
```

You can edit points and/or add a point by clicking on the plot with your mouse. To edit the currently selected point, click the left mouse button at the desired location. To add a new point, press the Ctrl key while clicking. If the *Snap mouse* box is checked, the coordinates of the point selected with the mouse will be rounded to "nicer" values: one tenth the tick mark interval on linear scales, for example. If you want to enter a new point that lies outside the extents of the graph, use the *Zoom out* button or one of the 4 pan buttons.

Scaling

The default scaling for any new XY plot is linear X, linear Y. You can change the scaling (logarithmic X and/or Y, probability scale, polar chart, bar chart, ternary plot, etc.) by right-clicking within the box surrounding the plot and selecting the scaling option from the drop-down menu. Scaling options are also accessible via the Linear/Log Scaling command on the *Options* menu.

Symbols/Line Types

To change the symbol type or line type used by a curve, right-click on the curve and select the

appropriate symbol and/or line type. Alternatively, select the <u>Symbol/Line Styles</u> command on the *Options* menu.

Scale Extents, Tick Marks, Plot Size

To change the extents of the X or Y axis, change the tick mark interval used, or to change the physical size of the plot, select Options>Extents/Intervals /Size.

Number Format

To change the number format used on any axis (to dates, date and time groups, multiples of PI, powers of 10, etc.), right-click on any of those numbers and select the appropriate format. To specify that a 1000 separator should be used, to specify the number of decimal places used, or to change the orientation of those numbers, right-click on any of those numbers and select "More number format options...". Alternatively select the <u>Number Formats</u> command on the *Options* menu.

1.2 Getting Started with XYZ Surface Plots



What is the source of your data?

A text file on disk

Excel or another program

An equation

No external source - enter data in DPlot

1.2.1 Getting Started with XYZ Surface Plots - Data From a Disk File

For tables of Z values, see below.

For 3 columns containing X,Y,Z data

Select the Open command on the File menu. Select file type K on the Select File Type dialog:



This format is an ASCII text file containing randomly-spaced 3D points, one X,Y,Z triplet per line. Comma-separated values (CSV) files produced by Microsoft Excel may be imported using this file type. Values may be separated by commas, spaces, or tabs.

A convex triangular mesh is generated, and each triangle in the mesh is considered planar when drawing contour levels. Areas outside the triangular mesh are not interpolated and not drawn. To delete extraneous triangles see the "How do I?" topic <u>"How do I force DPlot to create a triangular mesh of my 3D points that is not convex?"</u>. You can generate a smoother plot after reading one of these file with the Generate Mesh command on the Options menu.

For surface plots, points with identical X,Y coordinates are removed, preserving the point with the maximum Z value. This differs from 3D scatter plots, for which all points are preserved.

This file format may optionally contain 3 title lines and X, Y, and Z axis labels preceding the data. If included, these lines should start and end with double-quotation marks. DPlot reads labels in the order 1st title line, 2nd title line, 3rd title line, X axis label, Y axis label, Z axis label. So if you want, for example, to include all axis labels but have only one title line, the 2nd and 3rd lines in the file should be pairs of double-quotation marks. Alternatively, if the X, Y, and Z axis labels are on the same line (such that there are 3 pairs of double quotation marks on this line), DPlot will interpret this line correctly regardless of whether any title lines precede it.

For a table of Z values

If your data file consists of Z values only, with each line of the file containing Z values at a constant Y and each column corresponding to a different X, select the *Open* command on the *File* menu. Select file type L on the *Select File Type* dialog and the *Z Table* plugin module:

Select File Type	×
File Types	
🗇 A DPlot File	G Pacific digitizer, old format
○ B in (number of points)	 H Pacific digitizer, new format
x(1), y(1), , x(n), y(n)	I Nicolet time domain
○ C n dx (assumes x(1)=0) y(1),, y(n)	 J Hardened Data Acquisition System (HDAS)
D Multiple columns (2D) e.g. CSV files	⊘ K 3D or 4D - X,YZ, columns e.g. CSV files
E Unformatted binary	I Plugins
n (number of points) x(1), y(1), , x(n), y(n)	Z Table
🔘 F Pacific digitizer model 9820,	M 1D Statistics
Bendix/DNA format	
OK	Cancel Help

The Z Table plugin is an optional component; if you do not see this entry under Plugins you will need to reinstall DPlot to use it. This format is an ASCII text file containing Z values only. Each line of Z values in the file corresponds to a constant Y value. Likewise each column corresponds to a constant X. Comma-separated values (CSV) files produced by Microsoft Excel may be imported using this file type. Values may be separated by commas, spaces, or tabs.

After clicking OK and selecting a file you will be presented with a dialog box prompting you for the X and Y extents:

Z Table (Surface Plo	ot) Extents		? <mark>×</mark>								
This plugin expects to read tabulated z values from an ASCII text file in the form: z(x1,y1), z(x2,y1),, z(xn,y1) z(x1,y2), z(x2,y2),, z(xn,y2) OK Cancel											
z(x1,ym), z(x2,ym)	l,, z(xn,ym)										
where n is the nu values] and m is l values]. Rows ar to be evenly spac specified below. I separated by con											
Xmin: 🚺]	Ymin:	0									
Xmax: 30 Ymax: 30											

Rows and columns are assumed to be evenly-spaced between the specified extents.

Regardless of the file format, the surface will generally be presented initially as a 2D view with contour lines. To switch to 3D or for other rendering options, see the <u>Contour Options</u> command on the Options menu.

1.2.2 Getting Started with XYZ Surface Plots - Data From Another Program

Four methods are available for importing data for an XYZ surface plot from other Windows applications.

NOTE: For the *Paste* and *Insert Link* commands, if you are starting with a blank document then ensure that the blank document is of the correct type by selecting *New>3D surface plot* on the *File* menu. Otherwise DPlot will interpret the data as X,Y values.

The DPlot Interface Add-In for Microsoft Excel

This feature is of course only applicable if the source of your data is a Microsoft Excel worksheet. In general this method is superior to those described below. For more detailed information on Add-In commands see the Help command on the DPlot menu in Excel. For best results study the example spreadsheet <My Documents>\DPlot\examples.xls.

The XYZSurface command expects a selection of 3 columns with an optional 4th column containing data point labels:



The ZGrid command expects a table of Z values, with X values in the first row of the table, Y values in the first column.

	Α	В	С	D	E	F	G	Н	1	J	K	L	Μ	Ν	0	Р	Q	R	S	Т	U	V	W	Х	Y	Ζ
1	ZGrid test																									
2	Select B6:AG37 (or of course create your own table), then select ZGrid on the DPlot menu.																									
3	First row is X values, first column is Y. The remainder of the table is Z values at the corresponding X and Y.																									
4	It is not no	ecessa	ry for	the ro	ows ar	nd col	umns	to be	even	y-spac	ced, th	ough	they	are in	this e	examp	le.									
5																										
6		Y/X	-3.00	-2.80	-2.60	-2.40	-2.20	-2.00	-1.80	-1.60	-1.40	-1.20	-1.00	-0.80	-0.60	-0.40	-0.20	0.00	0.20	0.40	0.60	0.80	1.00	1.20	1.40	1.60
7		-3.00	0.14	0.33	0.51	0.67	0.80	0.90	0.96	0.99	0.98	0.92	0.83	0.71	0.56	0.39	0.20	0.00	-0.20	-0.39	-0.56	-0.71	-0.83	-0.92	-0.98	-0.99
8		-2.80	0.13	0.32	0.49	0.64	0.76	0.86	0.92	0.94	0.93	0.88	0.79	0.68	0.53	0.37	0.19	0.00	-0.19	-0.37	-0.53	-0.68	-0.79	-0.88	-0.93	-0.94
9		-2.60	0.12	0.29	0.44	0.58	0.69	0.78	0.83	0.86	0.84	0.80	0.72	0.61	0.48	0.33	0.17	0.00	-0.17	-0.33	-0.48	-0.61	-0.72	-0.80	-0.84	-0.86
10		-2.40	0.10	0.25	0.38	0.50	0.60	0.67	0.72	0.74	0.73	0.69	0.62	0.53	0.42	0.29	0.15	0.00	-0.15	-0.29	-0.42	-0.53	-0.62	-0.69	-0.73	-0.74
11		-2.20	0.08	0.20	0.30	0.40	0.48	0.54	0.57	0.59	0.58	0.55	0.50	0.42	0.33	0.23	0.12	0.00	-0.12	-0.23	-0.33	-0.42	-0.50	-0.55	-0.58	-0.59
12		-2.00	0.06	0.14	0.21	0.28	0.34	0.38	9 K	DPlot	- [exar	nples.	ds]	10										x	-0.41	-0.42
13		-1.80	0.03	0.08	0.12	0.15	0.18	0.21	9	File	Edit	Text	Ontio	ons G	enerate	e Info	Viev	v w	indow	Help	_				-0.22	-0.23
14		-1.60	0.00	0.01	0.02	0.02	0.02	0.03	9 4		Luit		oper							Theip	1				-0.03	-0.03
15		-1.40	-0.02	-0.06	-0.09	-0.11	-0.14	-0.15	-0		2 L	12		1P	> L			2	V	2	a _Z	K		~	0.17	0.17
16		-1.20	-0.05	-0.12	-0.19	-0.24	-0.29	-0.33	-9								<u> </u>				1				0.36	0.36
1/		-1.00 -0.08 -0.18 -0.28 -0.36 -0.44 -0.49 -0 examples.xls															0.53	0.54								
10		-0.80	-0.10	-0.23	-0.36	-0.47	-0.55	-0.63	-0				2			Z	Grid								0.69	0.70
19		-0.60	-0.12	-0.20	-0.43	0.00	-0.07	-0.75	-0				2.5	10	0.9	718				\boldsymbol{y}					0.01	0.02
20		-0.40	-0.13	-0.31	-0.47	-0.02	0.70	-0.04	-0				2.0	10	\leq	291	(Q	0.6-	0.10	s.	0.9				0.91	0.92
21		-0.20	-0.14	-0.33	-0.51	-0.00	-0.79	-0.09	-0				2	0.15	\geq	\geq		_			0.75				0.97	1.00
22		0.00	-0.14	-0.33	-0.52	0.00	-0.01	-0.91	-0				1.5		0	0.15	101	5	-0.2	0					0.99	0.00
23		0.20	-0.14	-0.33	-0.51	0.00	0.74	0.09	-0				1	131	-	21	17	0.45	5	\mathbf{N}	0.45				0.97	0.90
24		0.40	0.13	0.01	-0.47	0.62	0.67	0.75	-0				0.5	111	\sim	ંશા	1117	5	-	<u> </u>	0.15				0.91	0.92
20		0.00	0.12	-0.20	0.45	0.47	0.67	0.63	-0				> 0	-1118) } }]]		ĨI.	-)}		0				0.60	0.02
20		1.00	-0.08	-0.18	-0.30	-0.36	-0.00	-0.05	-0				-0.5	121	00-	7/1	1111	60		//	-0.15				0.03	0.54
28		1.00	-0.05	-0.12	-0.19	-0.24	-0.29	-0.33	-0				-1		\leq	0.45	11	S	0.6	21	-0.3				0.35	0.34
29		1.20	-0.02	-0.06	-0.09	-0.11	-0.14	-0.15	-0				-1.5	0.15				-			-0.6				0.17	0.17
30		1.60	0.00	0.01	0.02	0.02	0.02	0.03	d				-2	- 6	3-0.15		12	-	-0.15		-0.75				-0.03	-0.03
31		1.80	0.03	0.08	0.12	0.15	0.18	0.00	ď				-2.5	11		21		6	0.5	3	-0.9				-0.22	-0.23
32		2 00	0.06	0.14	0.21	0.28	0.34	0.38	d					S I	10.9	Nº1	Шú	11	391	11					-0.41	-0.42
33		2 20	0.08	0.20	0.30	0.40	0.48	0.54	d				-3	-3	-2	-1	0	1	2	3					-0.58	-0.59
34		2 40	0 10	0.25	0.38	0.50	0.60	0.67	d					-	-	-	x	-	-						-0.73	-0.74
35		2.60	0.12	0.29	0.44	0.58	0.69	0.78	0.00	0.00	0.04	0.00	0.17	0.01	0.40	0.55	0.17	0.00	-0.17	-0.32	-0.40	-0.01	-0.17	-0.00	-0.84	-0.86
36		2.80	0.13	0.32	0.49	0.64	0.76	0.86	0.92	0.94	0.93	0.88	0.79	0.68	0.53	0.37	0.19	0.00	-0.19	-0.37	-0.53	-0.68	-0.79	-0.88	-0.93	-0.94
37		3.00	0.14	0.33	0.51	0.67	0.80	0.90	0.96	0.99	0.98	0.92	0.83	0.71	0.56	0.39	0.20	0.00	-0.20	-0.39	-0.56	-0.71	-0.83	-0.92	-0.98	-0.99

For both the XYZSurface and ZGrid commands, the surface will generally be presented initially as a 2D view with contour lines. To switch to 3D or for other rendering options, see the <u>Contour Options</u> command on the Options menu.

The Paste command (Edit menu)

The Paste command is the method most likely to be supported by another Windows application. Once data is pasted from another application, DPlot owns the data. Changes to the data in the source application are not reflected in the DPlot document. As with the Paste Special and Insert Link commands, DPlot expects the data to be arranged in 3 columns: X, Y, and Z.

The Paste Special Link option (Edit menu)

The Link option establishes a link with the source of the data. Changes to the data made in the source application are reflected in the linked document. Unlike the Insert Link command, this method requires switching to the source application and back to DPlot.

The Insert Link command (Edit menu)

The Insert Link command performs roughly the same function as the Link option, except that it does not require switching to the source application to initiate the action.

1.2.3 Getting Started with XYZ Surface Plots - Data From an Equation

Use the Z=f(X,Y) command on the *Generate* menu to generate a contour plot with elevation (Z) a function of X and Y.

All of the function-parsing routines in DPlot have the same basic syntax, capabilities and limitations:

Available two-side-operators are +, -, *, /, ^. Available one-side operators: sin, cos, tan, asin, acos, atan, abs(absolute value), sqrt(square root), log, logh(or log10), exp (inverse natural logarithm), cosh,

sinh, tanh, rnd(random integer; argument is highest number that can be the output), fact(factorial), erf (error function), and cdf(cumulative distribution function). A complete list of available functions is shown <u>here</u>. These all take an argument that must be in parentheses.

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute "PI", "pi", or "Pi" in the function with "4*atan(1.0)"

NOTE: Be specific. DPlot does **not** interpret "2X" as "2*X".

Prevent divide by zero errors

If this box is **unchecked**, DPlot will trap divide by zero and many other math errors (e.g. logarithm of a non-positive value), report the value of x where the error occurred, and will not generate a curve. If this box is **checked**, when DPlot traps a math error it will shift x by a very small number and attempt the calculation again. This feature might be useful for generating curves of certain functions that would normally fail at specific values of x, e.g. sin(x)/x at x=0.

1.2.4 Getting Started with XYZ Surface Plots - Entering Data Values

If you are not already working with an existing 3D surface plot, first select *New>3D Surface* on the *File* menu. Select the *Edit data* command on the *Edit* menu to enter new data values or edit and/or delete existing values. For randomly-spaced 3D points, when you click OK DPlot will generate a new triangular mesh based on the existing data values.

Shortcut:

Click on the toolbar.

1.3 Getting Started with XYZ Scatter Plots



What is the source of your data?

A file on disk

Another program

An equation

No external source - enter data in DPlot

1.3.1 Getting Started with XYZ Scatter Plots - Data From a Disk File

Select the *Open* command on the *File* menu. The only built-in file format for 3D scatter plots is file type K on the *Select File Type* dialog:
Select File Type	×
File Types	
A DPlot File	🔘 G Pacific digitizer, old format
○ B in (number of points)	H Pacific digitizer, new format
x(1), y(1), , x(n), y(n)	I Nicolet time domain
(1) U n dx (assumes x(1)=0) y(1), , y(n)	J Hardened Data Acquisition System (HDAS)
D Multiple columns (2D) e.g. CSV files	♥ K 3D or 4D · X,Y Z, columns e.g. CSV files
E Unformatted binary	🔘 L Plugins
n (number of points) x(1), y(1), , x(n), y(n)	Z Table 🔹
 F Pacific digitizer model 9820, 	M 1D Statistics
Bendix/DNA format	
ОК	Cancel Help

Click OK. On the subsequent Open dialog box, be sure to check the Scatter plot box, else your data will be interpreted as a surface plot.

	sniines_error	_rand.csv	4/2//2009 9:09 AIVI	IVIICTOSOTE UTFICE E	AT KR
	🖳 square_test.	CSV	4/25/2009 12:45 PM	Microsoft Office E	1 KB
Documents	🐴 shlines_error	.csv	4/25/2009 7:39 AM	Microsoft Office E	91 KB
	🖾 ev09. nolabe	ls csv	4/12/2009 9·00 ΔM	Microsoft Office F	54 KR 🐣
	File <u>n</u> ame:	square_test	.csv	•	Open
Computer	Files of type: Comma-separated values (*.csv)		Cancel		
					<u>H</u> elp
Scatter plot	🔲 Pick Columns	to Plot			
~~~~		<u>4</u>	)		

This format is an ASCII text file containing randomly-spaced 3D points, one X,Y,Z triplet per line. Comma-separated values (CSV) files produced by Microsoft Excel may be imported using this file type. Columns may be separated by commas, spaces, or tabs.

Unlike 3D surface plots, all points are preserved in 3D scatter plots, including those with indentical X,Y values. Also, 3D scatter plots can contain multiple data sets. Data sets are separated by a single blank line. 3D scatter plots saved by DPlot as CSV files will use this same format.

This file format may optionally contain 3 title lines and X, Y, and Z axis labels preceding the data. If included, these lines should start and end with double-quotation marks. DPlot reads labels in the order 1st title line, 2nd title line, 3rd title line, X axis label, Y axis label, Z axis label. So if you want, for example, to include all axis labels but have only one title line, the 2nd and 3rd lines in the file should be pairs of double-quotation marks. Alternatively, if the X, Y, and Z axis labels are on the same line (such that there are 3 pairs of double quotation marks on this line), DPlot will interpret this line correctly regardless of whether any title lines precede it.

### 1.3.2 Getting Started with XYZ Scatter Plots - Data From Another Program

Four methods are available for importing data for an XYZ scatter plot from other Windows applications. **NOTE:** For the *Paste* and *Insert Link* commands, if you are starting with a blank document then ensure

that the blank document is of the correct type by selecting *New>3D* scatter plot on the *File* menu. Otherwise DPlot will interpret the data as X,Y values.

#### The DPlot Interface Add-In for Microsoft Excel

This feature is of course only applicable if the source of your data is a Microsoft Excel worksheet. In general this method is superior to those described below. For more detailed information on Add-In commands see the Help command on the DPlot menu in Excel. For best results study the example spreadsheet <My Documents>\DPlot\examples.xls.

The XYZScatter command expects 3 columns: X,Y,Z values. If the first row is non-numeric, it will be used as the axis labels.



#### The Paste command (Edit menu)

The Paste command is the method most likely to be supported by another Windows application. Once data is pasted from another application, DPlot owns the data. Changes to the data in the source application are not reflected in the DPlot document. As with the Paste Special and Insert Link commands, DPlot expects the data to be arranged in 3 columns: X, Y, and Z.

#### The Paste Special Link option (Edit menu)

The Link option establishes a link with the source of the data. Changes to the data made in the source application are reflected in the linked document. Unlike the Insert Link command, this method requires switching to the source application and back to DPlot.

#### The Insert Link command (Edit menu)

The Insert Link command performs roughly the same function as the Link option, except that it does not require switching to the source application to initiate the action.

#### 1.3.3 Getting Started with XYZ Scatter Plots - Data From an Equation

If you are not already working with an existing 3D scatter plot, first select New>3D Scatter on the File menu.

Use the  $\mathbb{Z}=f(X,Y)$  command on the Generate menu to generate Z values that are a function of X and Y.

Unlike 3D surface plots, you may use this command multiple times to produce multiple data sets in the same 3D scatter plot.

All of the function-parsing routines in DPlot have the same basic syntax, capabilities and limitations:

Available two-side-operators are +, -, *, /, ^. Available one-side operators: sin, cos, tan, asin, acos, atan, abs(absolute value), sqrt(square root), log, logh(or log10), exp (inverse natural logarithm), cosh, sinh, tanh, rnd(random integer; argument is highest number that can be the output), fact(factorial), erf (error function), and cdf(cumulative distribution function). A complete list of available functions is shown here. These all take an argument that must be in parentheses.

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute "PI", "pi", or "Pi" in the function with "4*atan(1.0)"

**NOTE:** Be specific. DPlot does **not** interpret "2X" as "2*X".

#### Prevent divide by zero errors

If this box is **unchecked**, DPlot will trap divide by zero and many other math errors (e.g. logarithm of a non-positive value), report the value of x or y where the error occurred, and will not generate a data set. If this box is **checked**, when DPlot traps a math error it will shift x or y by a very small number and attempt the calculation again. This feature might be useful for generating curves of certain functions that would normally fail at specific values of x or y, e.g. sin(x)/x at x=0.

### 1.3.4 Getting Started with XYZ Scatter Plots - Entering Data Values

If you are not already working with an existing 3D scatter plot, first select *New>3D Scatter* on the *File* menu. Select the *Edit data* command on the *Edit* menu to enter new data values or edit and/or delete existing values. You may edit an existing data group or create a new group.

#### Shortcut:

Click is on the toolbar.

## 1.4 Getting Started with 1D Plots



What is the source of your data?

A file on disk

Another program

No external source - enter data in DPlot

### 1.4.1 Getting Started with 1D Plots - Data From a Disk File

Select the Open command on the File menu. The only built-in file format for 1D plots is file type M on the Select File Type dialog:

Select File Type		×
File <u>T</u> ypes		
C 🛆 DPlot File	G B Pacific digitizer, old format	
○ <u>B</u> n (number of points)	○ <u>Н</u> Pacific digitizer, new format	
x(1), y(1), , x(n), y(n)	C I Nicolet time domain	
└─ <u>└</u> ─n dx (assumes x(1)=0) y(1),, y(n)	〇 J Hardened Data Acquisition System (HDAS)	
D Multiple columns (2D)	○ K 3D - X,YZ columns	
C E Unformatted binary	🔿 L Plugins	
n (number of points) x(1), y(1), , x(n), y(n)	WAV Audio	•
Pacific digitizer model 9820, Bendix/DNA format	• M 1D Statistics	
ОК	Cancel <u>H</u> elp	

This is an ASCII file containing one or more groups of amplitudes. These files produce a box-andwhisker plot or dot graph. The format for this file type is:

```
NumGroups
do i = 1,NumGroups
Label
NumPoints
do j = 1,NumPoints
Amplitude(j,I)
end do
end do
```

Amplitudes may be on separate lines or be grouped one or more values per line.

#### Example:

```
2
Group A
38
4.743281,4.781164,4.805654,4.83981,4.853127
4.86496, 4.875693, 4.8948, 4.903476, 4.911712
4.94844,4.955193,4.961815,4.968332,4.98113
4.987446,4.993731,5,5.006269,5.012554
5.018871, 5.03167, 5.038185, 5.044807, 5.051558
5.058462, 5.06555, 5.072855, 5.09652, 5.105203
5.114421, 5.124307, 5.13504, 5.146873, 5.17563
5.194346,5.218836,5.256719
Group B
49
3.973126, 4.124657, 4.222614, 4.297463, 4.359225
4.412506,4.45984,4.502771,4.542317,4.579189
4.613904, 4.646849, 4.678328, 4.70858, 4.7378
4.766151, 4.793769, 4.82077, 4.84726, 4.873327
```

4.899053,4.924515,4.949783,4.974923,5 5.025077,5.050217,5.075485,5.100947,5.126673 5.15274,5.17923,5.206231,5.233849,5.2622 5.29142,5.321672,5.353151,5.386096,5.420811 5.457683,5.497229,5.54016,5.587494,5.640775 5.702537,5.777386,5.875343,6.026874

produces this box-and-whisker plot:



You may then change the appearance of the plot using all of the Options menu commands.

### 1.4.2 Getting Started with 1D Plots - Data From Another Program

Four methods are available for importing data for a 1D plot from other Windows applications. **NOTE:** For the *Paste* and *Insert Link* commands, if you are starting with a blank document then ensure that the blank document is of the correct type by selecting *New>1D statistics plot* on the *File* menu. Otherwise DPlot will interpret the data as X,Y values.

#### The DPlot Interface Add-In for Microsoft Excel

This feature is of course only applicable if the source of your data is a Microsoft Excel worksheet. In general this method is superior to those described below. For more detailed information on Add-In commands see the Help command on the DPlot menu in Excel. For best results study the example spreadsheet <My Documents>\DPlot\examples.xls.

The Paste command (Edit menu)

The Paste command is the method most likely to be supported by another Windows application. Once data is pasted from another application, DPlot owns the data. Changes to the data in the source application are not reflected in the DPlot document.

#### The Paste Special Link option (Edit menu)

The Link option establishes a link with the source of the data. Changes to the data made in the source application are reflected in the linked document. Unlike the Insert Link command, this method requires switching to the source application and back to DPlot.

The Insert Link command (Edit menu)

The Insert Link command performs roughly the same function as the Link option, except that it does not require switching to the source application to initiate the action.

## 1.4.3 Getting Started with 1D Plots - Entering Data Values

If you are not already working with an existing 1D plot, first select *New>1D statistics* on the *File* menu. Select the *Edit data* command on the *Edit* menu to enter new data values or edit and/or delete existing values. You may edit an existing data group or create a new group.

### Shortcut:

Click on the toolbar.

© 2014 HydeSoft Computing, LLC



# 2 How do I ...?

Answers to common questions about DPlot.

#### Examples

How do I create the example plots?

#### **Data Entry**

How do I create a plot from scratch? How do I import data from another Windows application? How do I import data from a DOS application?

#### **Data Files**

How do I open a CSV (comma-separated values) file? How do I save a plot to an EPS (Encapsulated PostScript) file? How do I save a plot to a PDF (Portable Document Format) file? How do I save a collection of files to a different file format? How do I open and/or print a file from Explorer? How do I get rid of the prompt to save changed plots?

### Algebra

How do I graph an equation? How do I make a graph of a circle? How do I graph an inequality? How do I graph the solution to a system of inequalities?

#### Printing

How do I print colors and/or gray shades on a B&W printer? How do I print several documents unattended? How do I print multiple documents on the same page? How do I print blank graph paper?

#### Appearance

How do I change the direction of the coordinate axes?

How do I change the orientation of the numbers along each axis?

How do I create a series of plots with the same settings (scaling, labels, colors, etc.)?

How do I change the line widths used for the coordinate axes and/or the grid?

How do I change the format of the numbers appearing on the coordinate axes?

How do I force DPlot to create a triangular mesh of my 3D points that is not convex?

How do I create a scatter plot?

How do I replace the X (or Y) axis labels with custom text?

#### Scaling

How do I create a polar plot? How do I create a bar chart? How do I create a grain size distribution plot? How do I specify a tick mark interval? How do I force scaling on the X axis to be equal to the scaling on the Y axis?

#### **Startup Options**

How do I specify options on the DPlot command line?

#### Miscellaneous

How do I pan from side-to-side (or up-down) on a plot that I've zoomed in on? How do I temporarily hide one or more curves on a crowded document? How do I distribute my DPlot files to others?How do I get Help within DPlot?

## 2.1 Example plots

This section provides step-by-step procedures for recreating the example plots distributed with DPlot. (The example plots all have prefix of "ex" and an extension of ".grf". They are located in the DPlot folder on Windows 95 and 98 systems, and in My Documents\DPlot on all newer versions of Windows.)

Click on the corresponding picture for a full description of how that plot was created. You should start with the description of ex01.grf, as it contains quite a bit of information that is common to all example plots and is not repeated.







EX16.GRF Surface plot from a table of X,Y,Z values Non-convex surface plots

# 2.1.1 EX01.GRF - The basics



To begin, open the file ex01.grf. Select File>Open, select DPlot file, then look in the DPlot folder for ex01.grf. Alternatively, you can open ex01.grf by double-clicking on the filename in Explorer and/or by dragging the file from Explorer to the DPlot window.

In the Note in the lower right you'll notice that these curves are all based on an equation, namely:

 $\log_{10}(Cal/min/lb) = V/25 - 1.85$ 

Rearranging,

Cal/min/lb =  $10^{\sqrt{25-1.85}}$ 

So calories per hour per lb is given by:

Cal/hour/lb =  $60 \times 10^{\sqrt{25-1.85}}$ 

and the calories per hour for a specific rider weight is:

 $Cal/hour = W \times 60 \times 10^{\vee/25 - 1.85}$ 

Where W is the rider weight. We can use this equation to generate each of 9 curves shown in ex01.grf, substituting the actual rider weight for W. Select Generate>Y=f(X) and enter the values shown for "Y=", "From X", "To X" and "with interval dX" below (you can of course use whatever interval you'd like)

Y = f(x)	()		? 🗙
<u>E</u> nter f( Y= 1	OK		
Angles: radians radians degrees with interval gX= 0.5 ✓ Prevent divide by 0 errors			Cancel
Cor	nstants:		
A:	0	G: 0	
B:	0	Н: 0	
C:	0	I: 0	
D:	0	J: 0	
F:	0	К: 0	

Repeat the above for 120, 130, ..., 190 lbs.

#### **Attributes**

#### Line/Symbol Styles

If you aren't happy with the line styles or weights (or symbol styles) used for the curves, click the Line/Symbol Styles button and make the changes you want. For more information on this command click here. You can also change line and/or symbol styles by right-clicking on a curve.



Line/Symbol Styles button

### Colors

Colors can be changed by clicking on the Colors button (or Options>Colors>Edit) or by rightclicking on a curve.

Colors button

For more information on colors for XY plots click here.

#### Text

### Title and Axis Labels

Select *Text>Title/Axes* to enter the plot title and labels for the X and Y axes. If you mistype an entry after closing this dialog, just double-click on the label you want to change, or right-click on the label and select *Edit text*.

#### Legend

When using Y=f(X) or any of the Generate commands to produce a curve, the legend will automatically be assigned as the equation used to generate the curve. You can use the *Legend/Labels* command on the *Text* menu to change the legend. After closing this dialog, you can move the legend to the desired location by dragging it with your mouse. You can also double-click on a legend entry to edit the legend, or right-click on the legend to edit its attributes (framed, opaque, etc.)

#### <u>Note</u>

To enter a text notation, select <u>Text>Add/Edit Note</u>, or click the *Note* button.



Note button

Click the *Add Note* button to create a new note. Enter each line of the notation, pressing <Enter> at the end of each line. The only aspect of this note that might require a bit of explanation is the subscripted 10 for the log operator. There are a number of methods to add the subscript attribute; here is the easiest: Enter this line w/o the subscript:

log10(Cal/min/lb)=V/25 - 1.85

then use your mouse to highlight the 10 and click the subscript button.

x₂ subscript button.

This line of text will now look like:

 $\log{(d10)(Cal/min/lb)=V/25 - 1.85)}$ 

Close this dialog without checking *Button activated*. Why? You cannot move a button-activated note with your mouse. After using your mouse to drag the note to the desired location, if you want it to be button-activated right-click on it and select *Button activated*.

#### Scaling

The default scaling for any new XY plot is linear X, linear Y. You can change the scaling by rightclicking within the box surrounding the plot and selecting the scaling option from the drop-down menu. Due to the nature of the equation used to generate this plot, for example, linear X, logarithmic Y scaling will result in all curves appearing as straight lines. Scaling options are also accessible via the *Linear/Log Scaling* command on the *Options* menu.

#### **Plot Size**

The example is set up such that the box surrounding the plot is 7x7 inches (regardless of output device). Your plot probably fills the window at this point. If you want to specify a size, select *Options>Extents/Intervals/Size*, then check the *Specify plot size* box near the bottom of the dialog and enter whatever dimensions you'd like.

Alternatively, you can set the size of 2D plots by pointing the mouse cursor at a corner of the plot and dragging:



Dimensions will be shown in the units selected with the **Default dimensions** setting on the <u>General</u> <u>command</u> of the Options menu.

If you don't care for the default tick mark intervals, you can also change those in this dialog, as well.

#### Number Format

You will likely notice that the numbers along the Y axis ranging from 0 to 1700 or so do not use a 1000 separator. If you want to use a comma (or whatever character is specified as the 1000 separator with Control Panel) to delineate every 3rd digit, right-click on any number on the axis for which you want this change and select "More number format options..." and check the "Use 1000 separator" box. You can also right-click on those numbers to change the basic formatting of the numbers: powers of 10, engineering scales, multiples of PI, etc.

#### 2.1.2 EX02.GRF - Multiple dependent scales



This topic describes how to add X and Y axis labels to the top and right sides of a plot that are dependent on the labels on the bottom and left sides of the plot. This option is useful, for example, for displaying values in two different units.

If you already have an open plot that you would like to try this with, you are of course welcome to use it. Otherwise, open a new document window (select *File>New>XY plot*), return to this Help window and copy the table below (drag your mouse over the entire table including the headings, right-click and select *Copy*). Then select *Edit>Paste* in DPlot.

Striking

velocity, feet	/		
sec	Penetration	Perforation	No Spall
300	6.905	16.42	22.33
360	8.136	18.16	24
420	9.347	19.64	25.65
480	10.54	21.12	27.27
540	11.72	22.59	28.87
600	12.93	24.09	30.53
660	14.21	25.68	32.26
720	15.59	27.38	34.13
780	17.06	29.21	36.13
840	18.62	31.14	38.26
900	20.28	33.2	40.51
960	22.02	35.36	42.89
1020	23.86	37.64	45.38
1080	25.78	40.02	48
1140	27.79	42.52	50.73
1200	29.89	45.12	53.59

You'll notice that the headings are correctly interpreted as the X axis label for the first column and legend entries for columns 2 through 4. This works correctly in this case because the table is copied to the clipboard as HTML, so DPlot is able to easily determine where each column heading starts and stops. If the data is only copied to the clipboard as plain text, then column headings containing spaces and/or commas would only be interpreted correctly if they were separated by tab characters or if they

were surrounded by "quotation marks".

Pasting this data will create 3 new curves, with shared X values from the first column. In this example the amplitudes are in units of inches. So select *Text>Title/Axes* and enter "Inches" in the Y *Axis* box. Now... what if we would also like to display tick marks in centimeters? Select *Options>Multiple Axes*. Under Y *Axis Options* select  $Y2 = C + D^*Y1$ . This equation gives a relationship between the numbers on the two axes. One inch is equal to 2.54 centimeters, so in our case the appropriate entries are C=0, D=2.54. If you want to also display the X axis units in meters/second in addition to feet/second, under X *Axis Options* select  $X2 = A + B^*X1$  and enter 0 for A, 0.3048 for B (1 foot = 0.3048 meters).

Multiple Axes	
-X Axis <u>0</u> ptions: ○ Single X axis ○ X2 = A + B*X1	<u>A</u> : 0 <u>B</u> : 0.3048
Y Axis Options: Single Y axis Y2 = C + D*Y1 Independent Y2	<u>C</u> : 0 D: 2:54
ОК	Cancel <u>H</u> elp

Click OK, then Text>Title/Axes to enter "Centimeters" in the Alt. Y box and Striking Velocity, meters/ sec in the Alt. X box.

This feature can of course be used for other units. To display temperatures in degrees Celsius on the primary axis and degrees Fahrenheit on the secondary axis, for example, use C=32, D=1.8.

### 2.1.3 EX03.GRF - Polar plots and number formatting



In this example we want to create a polar chart. Start by generating a curve using Y=f(X). For our simple example we'll use the equation  $Y=abs(sin(3^*x))$  from X = 0 to 360 degrees with increment dX = 1 degree.

Y = f(X)	? 🛛
Enter f(x) (click "Help" for examp Y= abs(sin(3*x)) Angles: radians degrees with inter	le usage) (from X= 0 (to X= 360 (val dX= 1
Constants:	
A: 0	G: 0
B: 0	Н: 0
C: 0	I: 0
D: 0	J: 0
F: 0	К: 0

This gives a plot with linear X, linear Y scaling similar to this:



Right-click on the graph and select *Polar Coordinates*, or select <u>Options>Linear/Log Scaling>Polar</u> <u>Coordinates</u>. When using the Y=f(X) or similar commands, DPlot will automatically correctly set the scale for the angular values to either degrees or radians. If the source of your data is instead a text file or data pasted from the clipboard, then the angular values may be in the wrong units (radians are the default when DPlot is first installed). If the wrong units are assumed, the problem is usually obvious:



This is easily fixed. Right-click on the plot. At the bottom of the dropdown menu, select *Polar Plot Options*. (This option is also available on the main menu under *Options>Linear/Log Scaling*). Check *degrees* under *Angular Units*. Other formatting options specific to polar plots are also available on this dialog box.

Note that whether the angular input is in degrees or radians, the angular axis labels are in degrees. A handy option for normal linear scales with trigonometric functions or, in our case, polar coordinates is to instead display the angular labels using either multiples of PI or fractional values of PI. Right-click on any of the angular labels on the plot and select "Pi Fractions", or select the <u>Number Formats</u> command on the *Options* menu, ensure *X* axis is checked, and in the *Format* list select "Pi Fractions".

For more information on the Number Formats options, click here.

### 2.1.4 EX04.GRF - Multiple independent Y scales



Often you will run into a situation where you want to plot two curves on the same plot whose amplitudes are in different units. For example rainfall and corresponding flow rates in local tributaries, or velocity and acceleration, or in the case of this example, pressure and impulse. In such cases it is often useful to be able to associate these curves with different Y scales.

Open a new document window (*File>New>XY Plot*), then drag your mouse over the values below, rightclick and select Copy, then *Edit>Paste* into DPlot.

0.767	0
0.767	1002
0.8028	837
0.8385	698.6
0.8743	582.9
0.91	486.1
0.9457	405.2
0.9993	308
1.053	233.9
1.124	161.7
1.196	111.5
1.285	69.83
1.41	35.94
1.571	15.03
1.839	3.289
2.554	0

This gives a pressure vs. time waveform. Generate an impulse curve (area under the pressure curve) by clicking the *Integrate* button (or selecting *Generate>Integrate* a Curve).

### Integrate button

You'll notice that the amplitudes of the two curves are quite different. In this example this isn't critical, but in a plot with many more points and more detail we'd needlessly be giving up resolution. And besides, since the two curves have different units we cannot easily create a Y axis label that is meaningful. The solution is to associate the impulse curve with a different Y scale. Select *Options* >*Multiple Axes*. Under Y *Axis Options* choose *Independent Y2*. This will cause a list of curves to be displayed. You'll notice that the first curve is not included in the list. This is because the first curve in a plot must always be associated with the primary Y axis. Select "2: Curve #2" in the list, then Y2 in the axis box:

Multiple Axes	×
X Axis Options:	
Y Axis Options:	Curve: 2: Curve #2 Y1 Y2 Y2
For grid lines, use inde	pendent tick marks on dependent scales Cancel Help

Now, how to distinguish between the two curves? In other words, which curve is Pressure and which is Impulse? In the case of this example the answer might be obvious to most, but in other cases a visual cue would be very helpful. There are at least 3 ways to accomplish this:

1) With a legend. Select <u>Text>Legend/Labels</u> and enter the text you want as a descriptor for curve 1, then click the up arrow next to "Curve No." and repeat for curve 2.



2) With point labels. Select <u>Text>Label Points</u>, then add labels at the points of your choosing that describe each curve, as shown here:



3) With axis labels drawn in the same color as the corresponding curves. This method is obviously of limited usefulness if you have more than 2 curves and want more than 2 colors. In the example, the pressure curve is drawn in red and the impulse curve is drawn in blue. You can control the color of **both** axis labels with the <u>Fonts</u> command, but we want unique colors for each axis label. You can embed color codes into the text itself on the <u>Titles/Axes</u> dialog (as well as most other inputs for text entries). To change the color of all or a portion of a text entry, highlight the text that you want to change

the color of and click the Color button:

Titles/Ax	es	<b>x</b>
Title 1:	Hemispherical Surface Burst	ОК
Title 2:	1000 lb TNT, Standoff = 10 feet	Cancel
Title 3:		
Title 4:		Нер
X:	Time, msec	
Y:	{\c0000ffPressure, psi}	🔲 Horizontal
Y2:	Impulse, psi-msec	🔲 Top-down
	X ₂ X ² C B I U ‡2 W	

### 2.1.5 EX05.GRF - Tripartite plots



This topic describes how to create a tripartite plot. A tripartite plot (also known as "four coordinate paper") consists of a logarithmic scale on both the X and Y axes and two additional logarithmic scales that are derived from the X and Y axes by some fixed relationship. It is typically used in vibration analyses to determine the peak responses (acceleration, velocity, and displacement) of many different frequency damped, single degree of freedom systems exposed to support motions. Those support motions might be due to earthquakes, explosions, vehicle collisions, etc. Generally a tripartite plot used in this manner will consist of frequency in Hertz on the X axis, psuedo velocity (maximum relative displacement times frequency in radians) on the Y axis, displacement lines at 45 degrees to the X axis, and acceleration lines at 135 degrees to the X axis.

If you already have an open plot that you would like to try this with, you are of course welcome to use that data. Otherwise, open a new document window (select *File>New>XY plot*), return to this Help window and copy the table below (drag your mouse over the entire table including the headings, right-click and select *Copy*). Then select *Edit>Paste* in DPlot.

Frequency,	Undamped	5.00%	10.00%
Hz	-	damping	damping
0.333333	0.218286	0.214353	0.210019
0.35	0.228792	0.22449	0.219779
0.4	0.259943	0.254452	0.248546
0.45	0.290521	0.283721	0.276522
0.5	0.322559	0.312257	0.30368
0.55	0.352453	0.340024	0.329994
0.6	0.378262	0.366991	0.355446
0.7	0.488396	0.456718	0.427845
0.8	0.568285	0.522135	0.482474
0.9	0.601147	0.546691	0.502352

55

1 1 25	0.577336	0.567315	0.554449
1.25	1 333305	1 233605	1 142899
2	2 399888	2 06377	1 793627
25	5 359572	3 865065	2 977338
3	9.37866	5.774712	4.089978
3.5	6.586246	4.171359	3.056242
4	3.428447	2.579845	2.075591
4.5	2.5476	1.928202	1.601665
5	1.741	1.50927	1.32306
5.5	1.716137	1.321356	1.156819
6	1.333306	1.148157	1.003342
7	1.005915	0.856357	0.743198
8	0.872728	0.634849	0.590422
9	0.577468	0.538079	0.535213
10	0.65936	0.501782	0.49244
12.5	0.609857	0.433688	0.414941
15	0.491792	0.376587	0.360203
20	0.45/1/	0.295703	0.282438
25	0.353804	0.239117	0.228736
30	0.210000	0.197542	0.189377
30	0.172909	0.10004	0.109024
40 45	0.190044	0.141010	0.130302
	0.158425	0.123334	0.110003
55	0.118403	0 101482	0.094403
60	0 113993	0.092792	0.08651
70	0.111989	0.080279	0.074763
80	0.091674	0.070345	0.065573
90	0.084288	0.063053	0.058947
100	0.073453	0.057496	0.053711
125	0.063091	0.046978	0.044025
150	0.053322	0.038876	0.036598
200	0.035499	0.025902	0.02425
250	0.029394	0.022213	0.020661
300	0.023711	0.0172	0.016051
350	0.019072	0.012695	0.012572
400	0.018331	0.012627	0.011951
450	0.0163/5	0.012072	0.011298
511.5	0.014466	0.010823	0.009946

This table shows the peak velocity of a single-degree-of-freedom system due to a sinusoidal support motion for several <u>critical damping</u> ratios.

Initially you will see a standard linear X, linear Y plot with large peak values at a small (relative to the extents) frequency:

*r	_		-	-	-	-	-	-	-	-	_
	Ē	Unda 5.00 10.0	imped % dan 9% da	uping maina							
•											
2											
ļ,	-	100	144	200	280	300	360	400	40	800	880

To switch to a tripartite grid, right-click within the box surrounding the plot and select Tripartite Grid.

Other formatting options for this scale type are available on the right-click menu. In particular, *Square Log Cycles* will force the plot to be drawn such that the physical distance between cycles on the X axis

is identical to the physical distance between log cycles on the Y axis. This option is not shown if you have forced the plot to a specific size using *Options>Extents/Intervals/Size*.

By default DPlot assumes your input X values are frequency in Hertz, Y values are velocity in inches, and displays acceleration in g's and displacement in inches. Presets for other units are available with the *Tripartite Options* command on the right-click menu (or *Options>Linear/Log Scaling>Tripartite Options* on the main menu). For user-defined units not shown on that dialog, please see the Help topic for the *Tripartite Options* command.

### 2.1.6 EX08.GRF - Bar chart



A bar chart or bar graph is an XY plot with rectangular bars of lengths proportional to the value they represent. When you first create an XY plot the graph will always be initially drawn as a line graph with line segments between each data point. To switch to a bar chart, right-click within the box surrounding the plot and select "Bar Chart".

#### Example:

Select the data below, right-click and select Copy, then Paste into DPlot.

Floor-Load Data		
Live load, psf	Test 111902-01	Test 111902-02
0	10	13
20	17	15
40	41	34
60	42	37
80	35	40
100	19	26
120	22	20
140	16	18
160	6	8
180	7	6
200	2	1
220	3	2

As mentioned above, you will initially see a line graph with line segments between each data point:



Right-click within the box surrounding the plot and select "Bar Chart":



Floor-Load Data

Alternatively, select Linear/Log Scaling>Bar Chart on the Options menu. In either case this action will result in a bar chart with default settings:



To change bar chart-specific settings right-click within the plot again and select Bar Chart Options. Each of the options are thoroughly described in the <u>Bar Chart Options</u> topic.

### 2.1.7 EX09.GRF - Surface plot from an equation



This topic illustrates generating a 3D surface plot from an equation. First, open a new document window (*File>New>3D surface*). Now select *Generate>Z=f(X,Y)*, which allows us to generate a 3D surface with Z (height) values a function of X and Y. You are of course welcome to use any equation you like; this example uses Z=sin(x)*sin(x)*cos(y) with X and Y extents =  $\pm$  pi radians. Note that you can enter the value "PI" for 3.14159... in the extents boxes in addition to numbers and simple equations.

We've used an interval on both the X and Y axes of PI/30, which should be more than sufficient (in this case) to capture all of the details of the equation without trying our patience: That interval gives us 61 columns by 61 rows of data values for a total of 3721 points. On reasonably fast computers DPlot will take no more than a few seconds to plot 3D surfaces with 100,000 or so points.

Z = f(X, Y)							? ×
Enter f(x,y) (l	Click "Help" fo	r example	usage	)			ОК
Z= sin(x)*si	in(x)*cos(y)					•	Coursel
	Angles: • radians • degrees	from X= to X= dX=	-PI PI PI/30	from Y= to Y= ) dY=	-PI PI PI/30	] ]	Help
		P	revent	divide by 0 erro	rs		
Co	nstants:						
A:	0		G:	0			
B:	0		H:	0			
C:	0		l:	0			
D:	0		J:	0			
F:	0		K:	0			
Expand constants in legend/title							

This initially produces a two-dimensional plot with contour lines:



To switch to a 3D view, switch to shaded bands rather than contour lines, change the number of contour intervals or the upper and lower limits of those contours, and/or change the color scheme used for the contours, select *Options*><u>Contour Options</u>. The settings used for the example plot are shown here:

Contour Plot Options		₹ <mark>x</mark>
Type Shaded bands Contour lines Both lines and shades Waterfall plot Wireframe Bar chart 3D View in 3D Azimuth: 30 Elevation: 20 Perspective: 0 Preserve scale Sides	Input values: Minimum: -0.997736 Maximum: 0.9932181 Lines/Levels Results: Intervals: 20 + 1 0.9 Lower: -1 0.8 0.7 Upper: 1 0.6 Use actual displayed limits OR Custom levels >> Contour lines Major line width: 0.02'' Minor line width: 0.02'' Minor line width: 0.02'' Minor line width: 0.02'' Major lines every 1 + st line V Label every major line Transparent Black text Horizontal B/W based on luminance	Draw ✓ Axes ✓ Grid Borders Data points 0'' ← Cancel Apply ✓ Legend Label every: 1 → st interval Colors Color © Color © Gray scale © Custom → Axes: ▲ Background Borders: ↓ Window: ↓ ↓
Lighting: None Gouraud Phong Ambient fraction (0-1): 0	Bar chart Bar width as ratio of spacing X: 1 Y: 1 Scale factors X: Y: Z: 1 1 1	Other lines: Axis major lines: 0.02" Axis minor lines: 0.016" Wireframe lines: 0"

For a description of all of these options see the <u>Contour Options</u> topic.

To show the X and Y axis values as fractional values of pi rather than the default number format, rightclick on any number on the respective axis and select "Pi Fractions".

To change the extents of the axes and/or the tick mark interval used on each axis, select Options> Extents/Intervals/Size.

#### 2.1.8 **EX10.GRF - Point labels**



This topic illustrates DPlot's point labeling feature. First, open a new document window (File>New>XY Plot). Now we'll recreate the curves shown in the example. This example shows the motion with time of an object that is given an initial upward velocity and subjected only to gravity (drag and other forces are neglected). The equation of motion, repeated in the example's notation, is:

$$y = V_o t - gt^2/2$$

#### where

- y = height of object
- V_o = initial vertical velocity
- g = acceleration of gravity

t = time

We'll recreate the 3 curves shown, substituting our own values in the equation. We'll use US units (velocity in feet/sec, g in ft/sec^2) for this example.

Y=f(X)			? 🛛
Enter f(x) (click ' Y= A*x-G*x^2/2	'Help'' for example usage 2	)	
C C	ngles: radians degrees with inte	from X= 0 to X= 2*A/G rval <u>d</u> X= 0.04*A/G	Cancel
Cor	nstants:		
A:	150	G: 32.174	
B:	0	H: 0	
C:	0	l: 0	
D:	0	J: 0	
F:	0	К: 0	

We're making use of constants throughout the settings in this dialog: A for the initial velocity in feet/ second and G for the acceleration of gravity in ft/sec². Constants are useful when plotting the same equation multiple times, changing only one or two parameters, and also when using physical constants that don't change. The constants will retain their values until and unless we set them to some other values.

We start at x (time) = 0, but where to stop? We're only interested in plotting the motion until the object returns to the ground surface (y=0). There are two ways to go about this:

1) You can use a large time value such that the calculation will proceed past the time when the object returns to y=0, then use <u>Extents/Intervals/Size</u> to specify the extents of the plot. In our case we also want to find the maximum height of the object, so if using this approach we'd need to use a fairly small time interval to ensure that we find the maximum (or very close to it).

2) In many cases you can easily predict at what x (in this case time) the maximum amplitude will occur. For our equation of motion this is relatively easy: the maximum height (y) occurs where the velocity is 0:

$$y = V_0 t - gt^2/2$$
  

$$y' = V_0 - gt$$
  
with  $y' = 0, t = t_{max} = V_0/g$ 

Likewise, solving for y=0 we find the time at which the object returns to y=0 is  $2^*V/g$ , or twice the time to the maximum height. So we've used a "to X" value of 2V/g and a time increment that divides evenly into both the time of maximum height and the "to X" value, so we can be sure that we'll be calculating the true maximum height of the object.

Repeat the above for 100 and 50 fps...



...then use Extents/Intervals/Size to specify the plot extents: X from 0 to 10, Y from 0 to 450.

Now we want to label the points at which the maximum height occurs for each curve. Select *Text>Label Points*. Click *Add* to add a label. Select "Curve #1" in the *Curve* list (it should already be selected). We can scan through all points using the *Point* # scrollbar, but there is a much easier way: Under *Jump to*, click the *Ymax* button. This will automatically move the crosshairs and the scrollbar to the point with the maximum Y value. Now enter a label in the *Label* box... we could type in the actual amplitude, but again, there is an easier way. Type "\$Y ft". \$Y will be substituted with the actual Y value of the point associated with the label, \$X will do the same for the X value. One advantage to using this method is that the labels will be changed if the data is updated. Say, for example, that you multiply all Y values by 0.3048 to show height in meters rather than feet.

Repeat this procedure for the other two curves, and you should now have a plot that resembles the example. As with the legend and notations, you can move a point label by dragging it with your mouse, and change its attributes using the right-click menu.

### 2.1.9 EX14.GRF - Triangle plot



This topic describes how to create a triangle plot. A triangle plot (also known as a ternary plot), is a graph of 3 variables. It is most often used in geologic studies to show the relative compositions of soils and rocks, but it can be more generally applied to any system of three variables.

The proportions of the three variables plotted always sum to some constant. In DPlot that constant is always the value 100, or 1.0, depending on the <u>Triangle Plot options</u>. Because the 3 values for any point always sum to a constant, any one variable is not independent of the others so only two variables must be known.

Each vertex of the triangle represents a 100% proportion of the variable to the left, and 0% of the variable to the right side of the vertex. As a proportion increases in any one variable, the point representing that sample moves from the base to the opposite point of the triangle. In DPlot the X values are used for the horizontal scale, Y values are used for the right side of the triangle, and Z values (left side of the triangle) are forced to 100-X-Y or to 1.0-X-Y.

If you already have an open plot that you would like to try this with, you are of course welcome to use it. Otherwise, open a new document window (select *File>New>XY plot*), return to this Help window and copy the table below (drag your mouse over the entire table including the headings, right-click and select *Copy*). Then select *Edit>Paste* in DPlot. **Why only two columns, and not three?** Again, the three components plotted on a triangle plot must sum to a fixed number. Therefore the third component is redundant. If you paste 3 columns into DPlot you will get two separate curves, each sharing X values from the first column.

Sand	Clay
54.5	15.6
39.4	18.8
79.0	19.8
91.2	0.6
15.4	66.8
23.5	9.9
39.6	52.7
32.6	34.3
88.3	10.5
31.0	53.8
51.9	5.5
76.5	8.9
65.7	17.1
70.6	4.0
93.9	2.0
42.3	10.4
77.4	16.8
70.0	0.6
72 1	89

After selecting Paste in DPlot you will initially get a jumbled mess on a standard linear X, linear Y scale:



Right-click within the box surrounding the plot and select "Triangle plot". Alternatively, on the menu select Options>Linear/Log Scaling>Triangle plot.

You will likely notice that line segments between points have been turned off, and a symbol is drawn at each data point. Since triangle plots are generally used to plot discrete points rather than smooth curves, DPlot by default will always turn off line segments and, if no symbol is currently selected, turn on symbols. If this is not the behavior you desire (that is, you **want** line segments between data points), select *Options>General* and uncheck the *Always force symbols on/lines off for triangle plots* box. Whether this box is checked or not, you can always modify the line type and/or symbol type by right-clicking on a point and selecting *Line style* and/or *Symbol style*. Alternatively you can use *Options>Symbol/Line Styles* or click the *Line/Symbol Styles* button.

## Line/Symbol Styles button

In our example the **Sand** and **Clay** labels from our table are used for the X and Y axis labels. To add a label for the third axis, select *Text>Title/Axes* and enter the desired label in the *Z Axis* box. You can also bring up this dialog by double-clicking on any existing axis label.

If you want to change the tick mark interval, right-click on any of the numbers along each axis and select *Extents/Intervals/Size*. Select the *Specify interval* option, and enter the desired interval in the *Tick mark interval* box.

### **Special Purpose Plots**

DPlot includes several special purpose triangle plot types, accessible by right-clicking on the plot and selecting Triangle Plot Options. Currently DPlot will draw region boundaries and labels for:

Х

#### Plot type

Υ

Ζ

USDA Soil Classifications	Sand	Clay	Silt
IUGS Mafic Rock	Olivine	Plagioclase	Clinopyroxene
IUGS Ultramafic Rock	Clinopyroxene	Olivine	Othopyroxene
Ab-An-Or Mineral Classification	Orthoclase	Anorthite	Albite

If you would like to use one of these options but your data is not arranged in the proper order, you can perform an operation on the data to swap components. For example if your soil data has X=sand, Y=silt, use Y=100-(X+Y) with the *Operate on Y* command on the *Edit* menu.

### 2.1.10 EX16.GRF - Surface plot with concave boundary



The example plot shows contours of some phenomenon measured at various gauge locations in an Lshaped room. In this topic we'll recreate a less complex, but similar plot. The gauge locations for this exercise are shown here:

_	_	-	-	-					
[	-				I.				
۲.	•	•	•		1				
					٠				
	•								
[	_	-	-	-	1				
Γ.					Τ.				
					٩.				
•		•	•		٠				
L .					1				
Γ.	_				T				
r -					1				
•					٠			•	7
		•	•		•				÷
L				-	-		-	-	1
Г	-	-	-	1	7			1	1
			•		•				1
-	-								-

...and the peak measurements from those gauges are displayed in the following table. The first two columns are the X and Y coordinates (horizontal distance in a plan view) within the room; the third column is our "Z" values that we want contours of. Select the entire table with your mouse (click and drag over the table), right-click and select *Copy*. In DPlot, select *Paste Special* on the *Edit* menu, then check 3D - X, Y, Z values.

0.1	0.1	28.03
0.1	4	28.34
0.1	8	27.82
0.1	12	28.48
0.1	15.9	39.32
0.1	20	49.54
0.1	24	58.19
0.1	28	81.38
0.1	32	81.38
0.1	36	58.19
0.1	40	49.54
0.1	44	39.67
0.1	48	28.48
0.1	51.9	41.42
4	0.1	26.44
4	4	26.18
4	8	23.73
4	12	17.88
4	15.9	22.79

4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 8 8 8 8	$\begin{array}{c} 20\\ 24\\ 28\\ 32\\ 36\\ 40\\ 44\\ 8\\ 51.9\\ 0.1\\ 4\\ 8\\ 15.9\\ 20\\ 24\\ 28\\ 32\\ 36\\ 40\\ 44\\ 8\\ 51.9\\ 0.1\\ 4\\ 8\\ 12\\ 15.9\\ 20\\ 24\\ 28\\ 32\\ 36\\ \end{array}$	38.43 63.3 73.4 73.4 63.3 38.43 22.08 19.37 22.73 27.5 22.16 16.84 20.88 25.03 48.5 73.4 110.16 73.4 48.5 25.35 21.94 30.85 20.37 21.57 19.46 19.98 25.03 48.5 73.4 110.16 19.98 25.03 48.5 73.4 110.16 19.98 25.03 48.5 73.4 110.16 19.98 25.03 48.5 73.4 110.16 19.98 25.03 48.5 73.4 110.16 19.98 25.03 48.5 73.4 110.16 19.98 25.03 48.5 73.4 110.16 19.98 25.03 48.5 73.4 110.16 13.4
12 12 12 12 12 16 16 16 16 16 16 16 16	40 44 48 51.9 0.1 4 8 12 15.9 20 24 28 32	48.5 25.35 21.94 30.85 23.59 20.66 19.87 19.17 21.83 38.43 63.3 73.4 73.4
16 16 16 16 19.9 19.9 19.9 19.9 19.9 19.	36 40 44 51.9 0.1 4 8 12 15.9 20 24	63.3 38.43 22.08 20.05 22.73 19.15 19.81 20.76 18.03 39.32 49.54 58.19

19.9 19.9	28 32	81.38 81.38
19.9 10 0	36 40	58.19 49 54
19.9	44	39.67
19.9	48	28.48
19.9	51.9	41.42
24	0.1	21.69
24	4	21.53
24	8 10	21.04
24 24	12	20.39 17 <i>1</i>
24	0.1	23 17
28	4	20.61
28	8	17.92
28	12	20.06
28	15.9	22.22
32	0.1	28.97
১∠ ३२	4 8	23.43
32	12	19.9
32	15.9	22.33
36	0.1	25.44
36	4	24.93
36	8	22.55
36	12	21.01
30 0	15.9	23.54
39.9	4	27 66
39.9	8	25.67
39.9	12	24.79
39.9	15.9	29.21

The resulting plot will look something like this:



What happened to the L-shape? To help understand what happened here select the <u>Contour Options</u> command on the <u>Options</u> menu and check <u>Borders</u> under <u>Draw</u>. This will show the edges of the triangles that DPlot produced from our 3D points:



As shown above, the plot includes triangles outside our room that we don't want. Why are those

triangles there? DPlot **must** make an assumption about what our surface looks like, since our table of X, Y,Z points contains no connectivity information or any clues about which points define the outside edges of the surface. The assumption made by DPlot is that our surface is convex - it will never produce a triangular mesh with concave exterior angles or with interior holes. For a great many problems this will be the "correct" surface, but is clearly incorrect in this case. We can easily remedy this, though, using either of two methods:

#### **Delete Triangles**

Select the *Delete Triangles* command on the *Edit* menu and click on the triangles outside the L-shape. As you click on a triangle it will be redrawn in reverse video. If you select a triangle accidentally, click it again to toggle it back on.



There is one gotcha concerning the *Delete Triangles* command: if you edit one or more X or Y values of your plot in any way, DPlot will generate a new triangulation, which of course will produce a new convex mesh. So if your data requires any editing you should of course perform that editing **before** using *Delete Triangles*.

Once all of the triangles you want to delete are selected, click the *Delete* button. Your plot should now resemble this:



If you'd prefer shaded bands of color as in the example plot rather than contour lines, select the *Contour Options* command on the *Options* menu and select *Shaded bands* under *Type*.



#### **Define boundary**

For a more persistent change to the plot shape than *Delete Triangles* provides, use the <u>Define</u> <u>Boundary</u> command on the Options menu. This command allows you to specify the endpoints of the line segments that form the exterior boundary of your surface. (You can also use this command to specify **holes** in your surface; for more information see the Help topic for <u>Define Boundary</u>). Unlike *Delete Triangles*, triangles deleted by this command will not be regenerated every time a new triangular mesh is produced. Another advantage over *Delete Triangles*, though it may not be apparent in this example, is that the border may often be defined with many fewer points than there are triangles to delete. For example, in the plot EX16.GRF that this example is based on there were originally 55 excess triangles, while the border can be completely defined with 6 points.

The border may start at any location and proceed in either a clockwise or counterclockwise direction around the surface. Border points do not necessarily have to be on the surface. X and Y components of each point may be separated by commas, spaces, or tabs, one point per line. As you enter each line you will see the border being updated on the plot, drawn as a reverse video line.

Along with typing the values you can use the mouse to select points by left-clicking the desired location on the plot. Points selected with the mouse will be added at the current line in the text box if the caret is in the leftmost position; otherwise these points will be added/inserted on the line following the caret. For best results with the mouse you might prefer to check the *Snap to nearest data point* box. If that box is checked, then the X and Y values of the data point closest to the mouse position will be used, rather than the converted mouse coordinates. *Snap to nearest data point* has no effect on values you type in, nor will checking this box modify values that are already present, regardless of how they were entered.

After entering all border points and selecting *OK*, DPlot will delete any triangles whose average X and Y value is outside the border. The average value is used as a check simply because it is mathematically trivial to compute and it is guaranteed to be inside the triangle. Note that this command will **not** clip triangles which cross the border but whose average value is inside the border.

## 2.2 Data Entry

### 2.2.1 How do I create a plot from scratch?

Edit Data button

- 1) Open a new document window if necessary by selecting the appropriate New command on the File menu. For XY plots you may also open a new document window by pressing CTRL+N.
- 2) For XY plots the current maximum number of curves and maximum number of data points per curve is shown on the right side of the status bar at the lower edge of the DPlot window. If either of these values is insufficient for the new document, enter new values by selecting the Resize Arrays command on the File Menu or by double-clicking the "Max Size:" area of the status bar.
- 3) Select the Edit Data command on the Edit menu or click the Edit Data button on the toolbar.



### XY Plots Only

You can edit points and/or add a point by clicking on the plot with your mouse. To edit the currently selected point, click the left mouse button at the desired location. To add a new point, press the Ctrl key while clicking. If the *Snap mouse* box is checked, the coordinates of the point selected with the mouse will be rounded to "nicer" values: one tenth the tick mark interval on linear scales, for example. If you want to enter a new point that lies outside the extents of the graph, use the *Zoom out* button or one of the 4 pan buttons. The zoom and pan buttons may be disabled, dependent on the scaling type. For example, you cannot pan with a polar plot.

For XY plots you of course need at least 2 points to draw a line segment, though you can draw a symbol at a single point.

#### 3D/4D Plots

At least 3 non-collinear points (projected onto the XY plane) with at least 2 points having a variation in Z are required for a **surface** plot. DPlot will remove points as needed to eliminate pairs of points with identical X,Y values, preserving the point with the maximum Z value.

Scatter plots require at least 2 points with a variation in Z. Unlike surface plots, you can have

multiple points with identical X,Y values.

#### 1D Plots (Box-and-whisker plots, dot graphs)

1D plots require at least 2 amplitudes per group. Groups with less than 2 points are removed.

Click OK to accept your changes. To edit any of the existing data points or enter additional data points, create a new curve for an XY plot, group for a 1D plot, or new data set for a 3D/4D scatter plot, select the Edit Data command again.

### 2.2.2 How do I import data from another Windows application?

Four methods are available for importing data from other Windows applications.

The DPlot Interface Add-In for Microsoft Excel

This feature is of course only applicable if the source of your data is a Microsoft Excel worksheet. In general this method is superior to those described below.

#### The Paste command (Edit menu)

The Paste command is the method most likely to be supported by another Windows application. Once data is pasted from another application, DPlot owns the data. Changes to the data in the source application are not reflected in the DPlot document.

#### The Paste Special Link option (Edit menu)

The Link option establishes a link with the source of the data. Changes to the data made in the source application are reflected in the linked document. Unlike the Insert Link command, this method requires switching to the source application and back to DPlot.

#### The Insert Link command (Edit menu)

The Insert Link command performs roughly the same function as the Link option, except that it does not require switching to the source application to initiate the action.

### 2.2.3 How do I import data from a DOS application?

- 1) Click the application's Control-menu box or press ALT+SPACEBAR to open the Control menu.
- 2) From the Control menu, choose Edit.
- 3) From the Edit cascading menu, choose Mark.
- 4) Select the information you want to copy.
- 5) Click the application's Control-menu box or press ALT+SPACEBAR to open the Control menu.
- 6) From the Control menu, choose Edit.
- 7) From the Edit cascading menu, choose Copy.
- 8) Switch to DPlot.
- 9) Choose the Paste command from the Edit menu, or press CTRL+V.

See also Paste command

## 2.3 Data Files

### 2.3.1 How do I open a CSV (comma-separated values) file?

If your files are XY data with X values in the first column and all subsequent columns being Y values for separate curves, you can drag-and-drop your CSV files onto DPlot or use Explorer's "Open with..." right-click command and DPlot will correctly interpret them, including (if present) titles and axis labels

(for a single curve) or legend entries for multiple curves. CSV files may also be opened with the File>Open menu command, of course (select the "<u>D</u> Multiple columns (2D)" file type).

For more information on DPlot's support of CSV files, see the Help topic for Multiple columns files.

DPlot includes support for 3D (X,Y,Z) data (" $\underline{K}$  3D – X,Y,Z columns") and for both 2D and 3D data allows you to specify which columns to plot, which column is X, or to specify that the data is alternating X,Y columns rather than X,Y,Y,Y. "CSV" is actually a bit misleading in this case, as DPlot will also allow tab- or space-separated columns.

DPlot will also allow you to save your data to a CSV file that can be read directly by Microsoft Excel.

#### Labels

For X,Y data files containing 3 (and only 3) columns or for 3D data files containing 4 columns, you may specify that the last column contains point labels by selecting the "Pick Columns to Plot" checkbox on the Open dialog. For example this data:

0.0, 0.00000000, "X=\$X, Y=\$Y" 0.1, 0.309016994, 0.2, 0.587785252, 0.3, 0.809016994, 0.4, 0.951056516, 0.5, 1.00000000, "X=\$X, Y=\$Y" 0.6, 0.951056516, 0.7, 0.809016994, 0.8, 0.587785252, 0.9, 0.309016994, 1.0, 0.00000000, "X=\$X, Y=\$Y" 1.1, -0.309016994,1.2, -0.587785252,1.3, -0.809016994,1.4,-0.951056516, 1.5,-1.00000000, "X=\$X, Y=\$Y" 1.6, -0.951056516,1.7, -0.809016994,1.8,-0.587785252, 1.9, -0.309016994,2.0, 0.00000000, "X=\$X, Y=\$Y"

will produce this plot:



## 2.3.2 How do I save a plot to an EPS (Encapsulated PostScript) file?

DPlot has no direct support for EPS output. But **Windows** does, and it's fairly simple to make DPlot produce EPS files. If you do not already have a PostScript printer installed:

- 1) Start Control Panel.
- 2) Select Printers, then Add a Printer.
- 3) Click Next to cycle through the first couple of dialogs. You're now going to install a PostScript printer. It doesn't matter if you actually have a particular printer or not since you won't be printing to it. If you know you'll only want to save B&W output, any of the HP LaserJet PostScript printers are good choices. If you want color information saved to the file, of course select a color printer, e.g. the HP Color LaserJet 5/5M. Of course if you know what the final output destination for your EPS files is and that printer driver is available, then you should install that printer.
- 4) After selecting a printer click Next. Under Available ports select FILE:. This is not critical since the Print dialog in DPlot will give you the option of printing to a file, but it might save you a few headaches of attempting to print to a nonexistant printer later on.
- 5) Click Next to cycle through the remaining dialogs, and... you're done.

After selecting the Print command in DPlot, be sure to select the PostScript printer from the list of available printers. Click **Properties**, then... well then what you do will be largely dependent on what version of Windows you're running and which printer driver you're using. For the HP LaserJet family of printers on Windows XP:

Click the Advanced button.

Under Document Options>PostScript Options set **PostScript Output Option** to "Encapsulated PostScript (EPS)".

To ensure compatibility with the widest variety of applications, set **PostScript Language Level** to 1, rather than the default 2. (Microsoft Word, for example, will open but will not display an EPS file using level 2, but displays level 1 files just fine.)

Click OK twice to return to the main Print dialog.



Back on the main Print dialog, be sure to check **Print to file**. You'll be presented with a **Print to File** dialog that will allow you to specify the name of the output file. Don't worry if the default filename extension is ".prn"; just type the filename with an extension of ".eps".

### 2.3.3 How do I save a plot to a PDF (Portable Document Format) file?

As with EPS files, DPlot does not directly support output of the PDF format and the solution is very similar to the <u>EPS problem</u>. However, Windows isn't much help here and you'll need Adobe Acrobat or any other Adobe product that includes Acrobat Distiller or the Adobe PDF virtual printer driver. If you have Acrobat installed you should see **Acrobat Distiller** and/or **Adobe PDF** as one of the possible choices of printer in DPlot's **Print** dialog. Select it. You may also have an option of **Acrobat PDFWriter**. However, PDFWriter has problems with DPlot contour plots, while Distiller and Adobe PDF handle these plots perfectly.

**Note:** The Adobe printer drivers offer several methods for compressing bitmap images, with JPEG compression being the default. The JPEG method **does** compress the image very effectively, but the result is generally poor with DPlot-produced graphics. ZIP compression produces larger files but much better quality images.

If you do not have access to one of those Adobe products, there are several low-cost PDF virtual printer drivers available on the internet. The following vendors provide free trial versions of low-cost PDF printer drivers that have all been tested successfully with DPlot:

Pdf995 from Software995.

docuPrinter LT from <u>Neevia Technology</u>.

PDFcamp from veryPDF.

Win2PDF from Dane Prairie Systems.

If you have an active internet connection, you can see an example PDF file created with the Adobe PDF virtual printer driver distributed with Adobe Acrobat 7. Similar results are achieved with each of the products listed above.
# 2.3.4 How do I save a collection of files to a different file format?

You can create a macro to perform one or more operations on each file that matches a file specification:

- 1) Select the *Macro* command on the *File* menu.
- 2) Under **Macro Name**, enter a name for the macro you're about to create. The name should be something that you'll recognize a week from now when you want to perform this operation again.
- 3) Click Edit/Create. This will bring up the Macro Editor window.
- 4) Type the following, replacing **bold** entries with the appropriate values. Each line is explained below.

```
Directory("c:\mydata")
FileType(1)
ForFilesIn("*.grf")
FileSaveAs(4096,"")
FileClose()
NextFile
```

## Command Description

- Directory sets the default directory (folder). All subsequent file specifications will be relative to this path.
- FileType tells DPlot what format the input files are. File type codes are given <u>here</u>. In this example this command is redundant, since DPlot will quickly determine on its own that the .grf files are DPlot files.
- ForFilesIn allows you to loop through one or more operations on several files matching the given file specification. In the above example, DPlot will open (one at a time) all files with an extension of **.grf** in the folder **c:\mydata**. All ForFilesIn commands must have a matching NextFile command, which terminates the loop. Alternative to providing a file specification, you can tell DPlot to read a list of files from an ASCII text file by using an "@" prefix for the argument. For more information click here.
- FileSaveAs saves the open document to a new filename in a different format. In the context of a ForFilesIn loop, only the file extension is used. The **4096** in the above example tells DPlot to save the document as a Windows metafile, with the default extension of **.wmf**. For more examples click <u>here</u>.
- $\label{eq:FileClose} FileClose \qquad \mbox{Closes the file opened by the } ForFilesIn \ \mbox{command}.$
- NextFile Terminates the ForFilesIn loop.

5) Click the Run button in the Macro Editor, then close the Macro Editor.

The ForFilesIn command can of course be used for many other operations that you want to perform on multiple files. You might, for example, want to use it to loop through a list of files and change the line and symbol styles in each document before resaving the document. For a complete list of all valid macro commands click <u>here</u>.

# 2.3.5 How do I open and/or print a file from Explorer?

The DPlot setup program offered the option to automatically associate files with a .GRF extension with DPlot. If you selected that option, then to open a file with a .GRF extension while in Explorer you can either:

• Double-click on the selected filename, or

• Select the filename, click the right mouse button, then select Open from the menu.

Likewise, to print a file with a .GRF extension while in Explorer, select the filename, click the right mouse button, then choose Print.

Whether the GRF extension is associated with DPlot or not, you can select a file in Explorer (or any other drag-drop source), drag it outside of Explorer's window, and drop it onto DPlot's window. When you drop files from Explorer onto DPlot running either as an icon or in it's normal (restored) state, DPlot opens the file dropped onto it. Drag-and-drop works with other file formats, too. DPlot attempts to determine the file type as described under <u>Specifying a File on the Command Line</u>.

You can select several files at once using the CTRL key and drag them all with the mouse. Each file will be opened in a separate document window within DPlot.

See also: DPlot File

#### 2.3.6 How do I get rid of the prompt to save changed plots?

Normally DPlot will prompt you before exiting to ask whether you want to save the current plot if it has been edited in any way. If you find this feature to be annoying, select the **General** command on the **Options** menu, and uncheck the **Prompt to save changed plots** box.

## 2.4 Algebra

#### 2.4.1 How do I graph an equation?

Three different *Generate* menu commands allow you to generate a graph of an equation. Which command you should use is dependent on the form of your equation.

Y=f(X) produces X,Y pairs from an equation you specify between given lower and upper limits.

Since the function will only produce one unique Y value for every X, it is not possible to plot, for example, a circle or an ellipse using this command. For circles, ellipses, etc., use:

<u>X=f(T), Y=g(T)</u> generates X,Y pairs from parametrical equations in which X and Y are both functions of a third independent parameter T.

This command is particularly useful for producing graphs of equations that cannot be handled properly by Y=f(X). For example, although  $Y=sqrt(R^2-X^2)$  is the correct equation for a circle, use of this equation with Y=f(X) will only result in a semicircle (the positive Y half). But you can easily produce a circle with X=f(T), Y=g(T) by applying a bit of trigonometry. With T varying from 0 to 360 degrees,  $X=R^*cos(T)$ ,  $Y=R^*sin(T)$  defines a circle of radius R (substitute the actual radius for "R") centered at the origin. For a circle centered at (H,K), you'd use  $X=H+R^*cos(T)$  and  $Y=K+R^*sin(T)$ .

The X=f(T), Y=g(T) command is of course useful for other geometric shapes that have more than one Y value for each X. To plot an ellipse  $X^2/A^2 + Y^2/B^2 = 1$ , for example, use X=A*cos(T), Y=B*sin(T) with, again, T varying from 0 to 360 degrees.

You might also find this command useful for generating unevenly-spaced data points. For example if you want your points to be evenly spaced on a logarithmic scale, you can use  $X=10^{(T)}$ , Y=some function of  $(10^{(T)})$ .

 $\underline{Z=f(X,Y)}$  produces (initially) a 2D contour plot of Z values, where Z is a function of X and Y. Since you can only have one surface per plot, this command will automatically open a new document window if the currently active window already contains a plot.

You can subsequently change the format to a 3D surface plot with the Contour Options command on

the Options menu.

#### Scaling

For XY plots, DPlot initially uses as much of the display as possible for your graph. For this reason, the physical distance between equal values on the X and Y axes will generally not be the same. In other words, a circle produced with X=f(T), Y=g(T) will seldom be circular initially. You can force a 1:1 scaling by using the <u>Extents/Intervals/Size</u> command on the *Options* menu. Check both the **Specify extents** and **Specify plot size** boxes. Enter the desired extents on both the X and Y axes under **Specify extents**, then use proportional values for the plot width and height under **Specify plot size**. For example, if the X axis extends from -2 to 8 (10 units) and the Y axis extends from 4 to 8 (4 units), then the plot width should be 10/4 = 2.5 times the plot height. Conversely, if you want the plot width and height to be specific values (square, for example), then change the plot extents so that they are proportional to the width and height.

For 3D data, DPlot initially uses a scale factor of 1 for each of the X, Y, and Z axes. This means that 1 unit on the X axis will have the same physical length as 1 unit on the Y and Z axes. Depending on the extents and type of the input data this may not be appropriate. There are a couple of alternatives to this default behavior. If you check **Always auto-adjust scale factors on 3D plots** on the *Options>General* dialog, then DPlot will set initial scale factors such that the plot is square in the XY plane and the Z extents appear to be about 1/2 that of X and Y. This is a useful feature if you frequently plot data sets with values in different units. You can also manually enter scale factors for each of the X, Y, and Z axes using the <u>Contour Options</u> command on the *Options* menu.

## 2.4.2 How do I make a graph of a circle?

The <u>Y=f(X)</u> command will only generate data points for which there is one unique Y value for every X. So even though Y=<u>sqrt(R^2-X^2)</u> is the correct equation for a circle, use of this equation with Y=f(X) will only result in a semicircle (the positive Y half). But don't despair; generating a circle is no problem with <u>X=f(T)</u>, <u>Y=g(T)</u>. With T varying from 0 to 360 degrees, X=R*<u>cos(T)</u>, Y=R*<u>sin(T)</u> defines a circle of radius R (substitute the actual radius for "R") centered at the origin. Of course if you want a circle centered at (H,K), you'd use X=H+R*cos(T) and Y=K+R*sin(T).

Initially the physical dimensions of your circle will most likely not be circular, depending on the window size. That's simply because by default, DPlot uses as much of the display monitor as possible to plot your data. You can force circular circles by using the <u>Extents/Intervals/Size</u> command on the Options menu and specifying equal extents for the X and Y axes and equal physical dimensions (check "Specify plot size" and enter equal values for the plot width and height). Alternatively, after checking both the "Specify extents" and "Specify plot size" boxes, click the "Set X:Y=1:1" button.



Circle of radius 5 with center at (4,6)

The X=f(T), Y=g(T) command is of course useful for other geometric shapes that have more than one Y value for each X. To plot an ellipse  $X^2/A^2 + Y^2/B^2 = 1$ , for example, use X=A*cos(T), Y=B*sin(T) with, again, T varying from 0 to 360 degrees.

#### 2.4.3 How do I graph an inequality?

Rather than a line or a curve, the solution set for an inequality involving X and Y fills an area on the XY plane. An equation defines the boundary or edge of the set. Although DPlot will not directly solve for and plot inequalities, you can easily do so yourself in a few steps. First rearrange the inequality (if necessary) in the form  $Y < (or \le, \ge, >)$  f(X). Use the  $\underline{Y=f(X)}$  command on the Generate menu to plot the boundary of the inequality. If the inequality is < or > rather than  $\le$  or  $\ge$ , the usual practice is to plot this boundary using a dashed line. To change the line type of the boundary line or curve, right click anywhere on the boundary and select "Line style". Finally, use the Fill Between Curves command on the Options menu to fill in the area above or below the boundary, depending on whether the inequality is > (above the boundary) or < (below the boundary).

#### **Example: Graph the solution to** $y \le 2x-3$ **.**

1. Select  $\underline{Y=f(X)}$  on the Generate menu. In the "Y=" box type 2*x-3. (Note the multiplication sign; you **must** include all operators in your equation; in this case DPlot would **not** recognize 2x as 2*x. Enter a start and end value for X. In this example we've used  $-10 \le X \le 10$  with an interval of 1. (The interval is not important since we are only graphing a line; two points are sufficient.) That operation produces a graph similar to:



2. The Y=f(X) command will automatically produce a legend as shown in the graph above. If you do not want this legend, right click anywhere on the legend and select "Hide Legend".

3. If you would like the physical distance between X and Y units to be the same, select the <u>Extents/</u><u>Intervals/Size</u> command on the Options menu, check "Specify extents" and enter the extents you want, then check "Specify size" and enter width and height values that have the same proportions as the graph extents. In this example extents of <u>+</u>10 for both X and Y and width=height works well.

4. You can change the appearance of the coordinate axes by clicking any of the Grid/Box buttons on the toolbar or selecting the <u>Grid Lines or Box</u> command on the Options menu. And if you want the coordinate axes to pass through X=0 and Y=0 rather than being placed on the left and bottom sides of the graph, check the Axes at 0 option under Grid Lines or Box. The above changes result in:



5. Select <u>Fill Between Curves</u> on the Options menu. Although this command is primarily used to fill in areas between two different curves, it can also be used to fill the region between a curve or line and some X or Y reference line. In the "1st Curve" list select "2*x-3". For the "2nd Curve" entry, since our inequality is < we are interested in Y values **less than** the line we just graphed. So we select "Bottom"

from the list. This will result in the area between the lower X axis and the line  $y=2^{x}-3$  being filled. Select the fill style and color you want to use, then click OK.

6. Finally, you'll want to add a label to the graph describing the inequality. You could place the label in the graph title (<u>Title/Axes</u> command on the Text menu). In this example we'll add the label using a notation. Select <u>Add/Edit Note</u> on the Text menu, or click the Note button on the toolbar. Click the "Add note" button. In the "Note" box, type "y < 2x-3" then select the < character and click the underline button. Check the "Frame note" and "Opaque" boxes, then click OK. You can drag the note to the desired location with your mouse. The finished product:



#### **Example:** Graph the solution to 2x - 3y < 6.

1. This example requires an extra step since we first need the equation expressed as y as some function of x. First solve for y:

- 2x 3y < 6
- -3y < -2x + 6
- y > 2x/3 2

Note the flipped inequality in the last step: don't forget to flip the inequality if you change signs (multiply or divide by a negative number).

2. Proceed as in the previous example with the Y=f(X) command, this time with  $Y=2^{*}X/3-2$ :



3. Note that in this example the expression is "y greater than", not "y greater than or equal to" For strict inequalities the usual practice is to show the boundary as a dashed line rather than a solid line. To change line styles right click anywhere on the line and select "Line style". After this and other formatting changes similar to the first example, your graph will resemble this:



4. Once again, use the Fill Between Curves command on the Options menu to fill the region above the curve (**above** because Y > f(X)).



We have used linear functions in these examples for simplicity. But there is nothing restricting you from using more complex functions (polynomials, trigonometric functions, etc.) for this technique. One important difference with non-linear inequalities is that when using the  $\underline{Y=f(X)}$  command, the interval between X values now **is** important in order to get sufficient detail. In general you'll want to use an interval of the X extents divided by 1000, or smaller. For high-order polynomials a smaller interval may be needed.

If you would like to graph the solution to a set of two or more inequalities at once, see the <u>How do I</u> graph the solution to a system of inequalities? topic.

#### 2.4.4 How do I graph the solution to a system of inequalities?

If you are not familiar with graphing inequalities in DPlot, please take a look at the <u>How do I graph an</u> <u>inequality?</u> page before proceeding. It includes specifics on a few DPlot operations that we'll assume you're familiar with in this topic.

A "system" of inequalities is a set of inequalities, usually two or three, that you deal with all at once. The solution to the system of inequalities is simply the intersection of the solutions for each of the individual parts. The technique for solving the system is identical to that for an individual inequality, with one additional step at the end.

**Example:** Graph the solution to:

 $3x - 2y \le 15$  $x + 8y \le 40$  $x \ge 2$ st as with solvin

Just as with solving single inequalities, it is usually best to solve as many of the inequalities as possible for "y" on one side. Doing so gives us:

 $y \ge 1.5x - 7.5$  $y \le 5 - x/8$  $x \ge 2$ 

For the first two inequalities we can use the  $\underline{Y=f(X)}$  command on the Generate menu, as we did with single inequalities. Use  $Y=1.5^*X-7.5$  for the first (don't forget the multiplication symbol), and Y=5-X/8 for the second. For both inequalities extents of  $\pm 10$  will work fine. Again, the interval is not critical since we are dealing with straight lines.

But what about the third? We cannot use Y=f(X) because the inequality is independent of Y. There are

a couple of ways to handle this, both relatively simple:

1. Select Edit Data on the Edit menu (or click the "Edit data" button on the toolbar), select "New Curve", and enter two endpoints for a line segment with X=2. The Y values don't matter as long as the extents will cover the Y range we are interested in. The other two inequalities with  $X = \pm 10$  give us extents of roughly  $-25 \le Y \le 10$ , so the two points (2,-25) and (2,10) will work nicely. Click OK after entering those two values.

2. Switch Y for X, use  $\underline{Y=f(X)}$  with Y=2 and -25  $\leq X \leq 10$ , then swap the X and Y coordinates of this line with the Swap X, Y command on the Edit menu.

After setting the extents of the graph as before (using <u>Extents/Intervals/Size</u> on the Options menu) you should now have a graph that resembles the following:



Now select the <u>Fill Between Curves</u> command on the Options menu. We want a fill area for each of the inequalities. For Fill No. 1 select  $1.5^*X - 7.5$  in the "Curve1" list. Since the inequality is  $\geq$  in this case we want to fill the area above the line, so in the "Curve2" box select *Top*. Likewise for the second inequality ( $\leq$ ) select *Bottom*, and for the third ( $X \geq 2$ ), select *Right*. The solution for each individual inequality is shown below:





The combination of all 3 solutions is a bit busy and not especially useful:



The solution we are looking for is the intersection of the three individual solutions. Return to the Fill Between Curves command and check the *Draw only the intersections of all areas* box. The solution:



Note that the fill color and style from the first fill area will always be used when this option is selected; all other color and style settings are ignored.

It is possible, of course, that there will be no solution to your set of inequalities. For example if you check the *Draw only the intersections of all areas* box for this graph:



no fill area will be drawn, since the two areas do not overlap (and since the boundaries are parallel, they will not overlap at **any** X). This is the correct solution, and DPlot will not issue any error messages or warnings in this case.

As with single inequalities, we have used linear functions in all examples for simplicity. But there is nothing restricting you from using more complex functions (polynomials, trigonometric functions, etc.) for this technique. One important difference with non-linear inequalities is that when using the  $\underline{Y=f(X)}$  command, the interval between X values now **is** important in order to get sufficient detail. In general you'll want to use an interval of the X extents divided by 1000, or smaller. For high-order polynomials a smaller interval may be needed.

## 2.4.5 How do I find the area under a curve?

To display the value of the area under a curve, use the \$AREA <u>shortcut</u> in any text entry for a plot: a title line, the legend, and/or a notation. When the graph is drawn, \$AREA will be replaced with the numeric value. The value will be displayed using the <u>number format</u> for the Y axis.

Use upper case letters for \$AREA; e.g. \$Area or \$area will be ignored.

The area is calculated using the trapezoidal rule:

$$A = \frac{1}{2} \sum_{j=2}^{n} (y_j + y_{j-1}) \times (x_j - x_{j-1})$$

If your document contains more than one curve, add a numeric suffix to \$AREA to indicate which curve you want the area under. For example \$AREA(3) will be replaced with the area under the 3rd curve. If that suffix is omitted in the legend, then \$AREA corresponds to the curve associated with that legend entry. For example \$AREA in the legend entry for the second curve is equivalent to \$AREA(2).

#### Example:

with the title of the graph set to "Area under curve=\$AREA":



## Area under curve=263.3333

The exact area under the curve

$$Y = 3 + 8X - \frac{1}{2}X^2$$

is given by:

$$Area = \int_{0}^{10} \left( 3 + 8X - \frac{1}{2}X^{2} \right) dX$$
$$= \left( 3X + 4X^{2} - \frac{1}{6}X^{3} \right) \Big|_{0}^{10}$$
$$= 3(10) + 4(10)^{2} - \frac{1}{6}(10)^{3}$$
$$= 263.33$$

which matches the value replacing the \$AREA shortcut. Keep in mind that the trapezoidal rule's accuracy is dependent on the spacing between points; the smaller the spacing, the more closely the trapezoidal rule will come to the rigorous solution.

\$AREA may also be used to find an enclosed area, if your curve forms a closed loop. If the curve has a clockwise orientation then \$AREA will result in a positive value; if the curve has a counterclockwise orientation then \$AREA will be negative. In the following example 3 circles with radius 5 were generated with the X=f(T), Y=g(T) menu command with the spacing in T set to 10, 1, and 0.1 degrees, respectively.





\$NPTS points, sqrt(Area/{\sp})=\$=(sqrt(\$AREA/PI))

# 2.5 Printing

#### 2.5.1 How do I print colors and/or gray shades on a B/W printer?

Windows simulates colors and gray shades on a B&W printer using patterns of black dots. For this feature to work as expected, the width used for a given line or curve must be greater than 1 printer dot. For best results when printing colors or gray shades, the minimum line width used for drawing curves should be about 12 dots, or 0.02 inches on a 600 dpi printer or 0.04 inches on a 300 dpi printer. Line widths used for each curve may be changed with the <u>Symbol/Line Styles</u> command on the Options menu, or by selecting the Symbol/Line Styles tool.

```
....
```

Symbol/Line Styles tool.

### 2.5.2 How do I print several documents unattended?

Use the <u>Macro</u> command on the File menu to create a macro for printing a series of files unattended. After creating and saving the macro, select the Run button from the Macro dialog box to process the commands. For example, the following macro commands will cause DPlot to open, print, then close all DPlot files in the directory C:\MYDATA:

```
Directory("C:\MYDATA")
FileType(1)
ForFilesIn("*.GRF")
   FilePrint()
   FileClose()
NextFile
```

NOTE: DPlot uses the default printer settings (set with Control Panel) to run macro commands. Changes made to the printer settings within DPlot will have **no** effect on the printer settings used by macros.

#### 2.5.3 How do I print multiple documents on the same page?

- 1) Open or create all documents that you want to print on the same page.
- 2) Select the Print command from the File menu.
- 3) Click the "Multiple" button in the Print dialog box, or press ALT+M.
- 4) Select the layout for the page: Vertical, Horizontal, or Two Columns.
- 5) Placement of documents on the page is controlled by the order in which the documents are selected. Select/deselect documents by clicking on the corresponding checkbox.
- 6) Enter the desired spacing between plots in the "Space between" box.
- 7) Click the "OK" button, or press ENTER.
- 8) Modify the printer settings if necessary, then click the "OK" button or press ENTER.

#### 2.5.4 How do I print blank graph paper?

This topic will walk you through creating publication-quality blank graph paper using a macro. You can easily adapt the macro to other scale types (this example uses log-log scaling) or other parameters for this same scaling. Don't want to be bothered with creating a macro? There really isn't much to it and the macro will make this operation easily repeatable, but in any case you can manually perform the same actions taken by the macro; see below for an explanation of each command and the equivalent menu

command or other action in blue text.

Several additional graph paper examples, with ready-to-print PDF files, are shown on the DPlot web site at http://www.dplot.com/graphpaper.htm.

- Select the Macro command on the File menu. Enter a name for the macro that you're about to create that you'll remember next week. In the example below we'll be creating log-log graph paper with 3 cycles on the X axis, 4 on the Y, so name the macro "loglog3x4". Now click the "Edit/Create" button.
- 2) Select the bold text below with your mouse, right-click and select Copy. In the Macro Editor, click the Paste button. If you want to assign a shortcut key to the macro, select one of the key combinations in the "Shortcut" list. Now click the "Save" button to save the macro. To now run the macro and create your graph paper, if you used a shortcut key you can simply press that key combination. Otherwise, select the Macro command from the File menu, select your macro from the list, and click "Run"

```
FileNew()
                       ' Create a blank XY plot
XYXY(2,1,1,1000,10000) ' Adds 2 points
Scale(4)
                       ' Log X - Log Y
LineType(1,0)
                       ' Don't draw the "curve"
SymbolType(1,0)
Size(1,6,8,1)
                       ' Size of the graph (6" wide x 8" high)
                       ' 12="none" for number format
NumberFormat(0,12)
NumberFormat(1,12)
                           This means numbers won't
                           be drawn on the axes
                       ' Pen for axes is 0.02 inches
LineWidth(-1,20)
LineWidth(0,10)
                       ' Pen for major grid lines is 0.01 inches
                       ' Pen for minor grid lines is 0.005 inches
LineWidth(-3,5)
GridLines(1)
                       ' Box around plot w/ grid lines
DenseGridLines(1)
                       ' Looks a bit crowded on the display,
                           but nice when printed
                       ' Set grid color to a med. gray
Color(0,128,128,128)
```

3) For help on any macro commands, simply move the text insertion point to the appropriate line and click the Help button in the macro editor. Each of the commands shown above is explained in more detail below. Again, blue text is used to describe the manual action that is equivalent to the macro command.

XYXY(2,1,1,1000,10000) ' Adds 2 points

DPlot will not display a graph (or blank graph paper) without data. So we create a new curve with 2 points, from (1,1) to (1000,10000). This gives us 3 log cycles on the X axis, 4 cycles on the Y axis. The resulting line will subsequently be hidden.

Select the Edit Data command on the Edit menu, or click the Edit Data button. Enter 1,1 for the first point. Click the Add button. Enter 1000,10000 for the second point. Click OK.

Scale(4)

' Log X - Log Y

With the text cursor on this line, click the Help button to show codes for other scale types. Right-click within the plot and select "Logarithmic X – Logarithmic Y".

LineType(1,0)

' Don't draw the "curve"

SymbolType(1,0)

We don't want to display the curve; we only created it to set our extents. So we use "None" for both the line type and symbol type.

Right-click on the line and select Line style>None and, if necessary, repeat with Symbol style>None.

size(1,6,8,1) ' Size of the graph
The first '1' simply says: "specify the size", rather than allow DPlot to fill the output device with the
graph. We've set the width of the graph box to 6 inches and the height to 8 inches. Note that this is the

same aspect ratio as the number of log cycles, so our log cycles will be square. If you use more or fewer log cycles then you'll of course need to adjust the size if you want to maintain square log cycles. Select the Extents/Intervals/Size command on the Options menu, check "Specify size", then enter the desired graph paper width and height in inches.

```
NumberFormat(0,12)
                             ' 12="none" for number format
NumberFormat(1,12)
                                 This means numbers won't
                                 be drawn on the axes
Click the Help button to show other number format codes. We've selected "None", so numbers won't be
drawn on the graph paper at all. All we'll have is the grid. If you want numbers to be printed, then of
course you should use a different number format or delete these lines entirely.
Right-click on any number on the X axis and select "None". Repeat for the Y axis.
LineWidth(-1,20)
                             ' Pen for axes is 0.02 inches
LineWidth(0,10)
                             ' Pen for major grid lines is 0.01 inches
LineWidth(-3,5)
                            ' Pen for minor grid lines is 0.005 inches
Pen widths are in 1000'ths of inches. These settings may require a bit of experimentation on your
printer.
Select the Symbol/Line Styles command on the Options menu. Under "Other lines" enter the desired
line widths for the axes (the box around the graph) and the major and minor grid lines.
GridLines(1)
                             ' Box around plot w/ grid lines
0=axes only, no grid lines, 2=box around plot with tick marks
Click the "Grid lines" button or select Options>Grid Lines or Box>Grid lines.
DenseGridLines(1)
                             ' Looks a bit crowded on the display,
                                 but nice when printed
Self-explanatory. Change the "1" to a "0" to see the difference.
Select Options>Extents/Intervals/Size. Enter any number greater than 1 for the "Minor grid/tick
divisions" on both the X and Y axes.
Color(0, 128, 128, 128)
                            ' Set grid color to a med. gray
```

By default, grid lines are drawn with a lighter gray (192,192,192). Click the Colors button.

# 2.6 Appearance

#### 2.6.1 How do I change the direction of the coordinate axes?

Normally, values on a linear or logarithmic scale increase from left to right (or bottom to top). You can change the direction by selecting the <u>Extents/Intervals/Size</u> command on the Options menu and checking the appropriate "Invert" checkbox below "Specify Extents". For example, if your logarithmic scale on the X axis normally extends from 1 on the left to 1000 on the right, check "Invert X" to change the extents to 1000 on the left, 1 on the right.

For 3D data you can achieve similar results by specifying negative scale factors for the appropriate axes with the <u>Contour Options</u> (for surface plots) or <u>Scatter Plot Options</u> (for 3D scatter plots) commands under the Options menu.

## 2.6.2 How do I change the orientation of the numbers along each axis?

Select the Number Formats command on the Options menu. Select the axis (X, Y, or Z) that you want to change the orientation for in the *Axis* box. At the bottom of this dialog, select *Horizontal*, 45°, or *Vertical* under *Orientation*. You can also use the <u>LabelDirection</u> macro command to set these directions.

These settings are ignored for polar plots, triangle plots, and 3D plots. For polar plots, you can

change the orientation of the radial labels to vertical with Options>Linear/Log Scaling>Polar Plot Options.

#### 2.6.3 How do I draw a smooth curve through sparse data?

For XY plots DPlot will always draw straight line segments between data points. The sawtooth curve below, for example, contains only 13 points, with an increment in X of 0.5:



If you would prefer a smooth curve through those points... you need more points. If you have additional data available then the best solution is of course to use that data. But you can also have DPlot generate additional data using one of several interpolation schemes with the <u>Equal Intervals</u> command on the Edit menu:



For a smooth curve you'll most likely want 100 or more points. In this example we'll use an interval of 0.05, which will produce 121 points between X=0 and X=6. The *Linear* interpolation method is useful for producing evenly-spaced points from unevenly-spaced data, but, as you'd expect, the result will appear identical to the original data. The *Quadratic* and *Cubic* interpolators will produce a smoother curve, but with discontinuities in the slope near either end of the curve. Generally the *Cubic Spline* interpolator is the best choice when all you're trying to do is draw a smooth curve. The result of *Cubic Spline* with an interval of 0.05 on the orginal data is shown below.



# 2.6.4 How do I create a series of plots with the same settings (scaling, labels, colors, etc.)?

There are a number of ways to accomplish this. Which method is best for you will depend on the frequency with which you use these same settings.

Create the first plot with all of the formatting that you want to use in subsequent plots. Select <u>Save</u> <u>Preferences</u> on the *File* menu. After creating any subsequent plots, whether by DDE, reading a file, pasting from the Clipboard, or entering values using <u>Edit Data</u>, use <u>Get Preferences</u> on the *File* menu to restore your previously saved preferences. This includes titles and axis labels (existing labels will not be replaced), colors, line styles, scaling, etc.

You can easily help automate this last step by creating a macro to open this preferences file and assigning a hotkey to the macro. For example, this macro:

```
GetPreferences("c:\program files\dplot\myprefs.set")
ViewRedraw()
```

will read the file myprefs.set and redraw the plot using the settings found in that file.

- 2) Same as above, but save your preferences to DEFAULT.SET in the default preferences folder (on Windows 95 & 98, the folder where dplot.exe is located. On Windows 2000 and newer, in % APPDATA%\DPlot\Preferences). DEFAULT.SET will be processed every time a new plot is created, without reading settings using the <u>Get Preferences</u> command.
- 3) If you want to restore some, but not all, of the settings saved to a preferences file, e.g. colors only, you can fairly easily create a macro to reproduce all the settings you're interested in. For more information on creating macros see <u>Creating and/or running a macro</u>.

## 2.6.5 How do I change the line widths used for the coordinate axes and/or the grid?

Line widths used for the coordinate axes (or box around the plot) and for the grid lines may be changed with the <u>Symbol/Line Styles</u> command on the Options menu, or by selecting the Symbol/Line Styles tool.

Symbol/Line Styles tool.

You can also select from 4 or 5 discrete line widths by right-clicking anywhere along a curve and selecting "Line Width".

## 2.6.6 How do I change the format of the numbers appearing on the X and Y axes?

The <u>Number Formats</u> command on the Options menu allows you to change the appearance of numbers along the X and Y axes. Available options include scientific notation (e.g. 1.454E+3), calendar dates, time of day, currency symbols, fractions, and multiples of .

These number formatting options can also be set by right-clicking any numbers along the coordinate axes.



# 2.6.7 How do I force DPlot to create a triangular mesh of my 3D points that is not convex?

The short answer is... you can't, at least not initially. DPlot is only reading X,Y,Z values from your input source, whether that source is a file, the Windows Clipboard, communication from another program, or the <u>Edit Data</u> command. This is not enough information to determine where the edges of the surface should be, or for that matter to determine whether your surface should have any interior holes. DPlot always creates a convex triangular mesh from X,Y,Z input, which for many applications is exactly what you want. It is of course possible that you will **not** want a convex mesh. In this case DPlot will have created triangles outside the desired limits of your plot. You have two options:

#### **Delete Triangles**

You can easily delete specific triangles to form the shape you want. View your data in 2D (rather than 3D) and for clarity ensure that *Draw Borders* is checked on the *Contour Options* dialog. Now select the *Delete Triangles* command on the Edit menu. Click on each triangle you want to delete. Triangles will be drawn in reverse video to indicate that they are selected for deletion. Click again on a triangle to deselect it. Once you've selected all of the triangles you want to

delete, click the *Delete* button, or press *Esc* or click the *Cancel* button to abort the operation. Depending on the complexity of your grid you may need to first zoom in to the area of interest. You can undo these deletions with the *Undo* command on the *Edit* menu.

**NOTE:** If you subsequently use the *Edit Data* command to edit any existing points, delete existing points, or add new points, DPlot will re-triangulate a new convex mesh. If you save a DPlot file after deleting triangles, that information is preserved. However, if you save the data as *Comma-separated values*, when you re-read this file DPlot will again triangulate a convex mesh.

#### **Define Boundary**

For a more persistent change to the plot shape than *Delete Triangles* provides, use the <u>Define</u> <u>Boundary</u> command on the Options menu. This command allows you to specify the endpoints of the line segments that form the exterior boundary of your surface. (You can also use this command to specify **holes** in your surface; for more information see the Help topic for <u>Define</u> <u>Boundary</u>). Unlike *Delete Triangles*, triangles deleted by this command will not be regenerated every time a new triangular mesh is produced. Another advantage over *Delete Triangles* is that the border may often be defined with many fewer points than there are triangles to delete.

#### 2.6.8 How do I create a scatter plot?

In this context, by "scatter plot" we mean a plot with symbols at each data point, and no line segments connecting the data points.

Right-click on a curve and select *Symbol style*, then select the desired symbol. Right-click again and select *Line style>None*. Note that you will want to select the symbol style **before** setting the line style to None, else the curve will not be visible.

Alternatively, change the line type for each curve to None and pick the desired symbol type using the <u>Symbol/Line Styles</u> command on the Options menu. The Symbol/Line Styles dialog includes many additional settings, including line weight, symbol size, symbol frequency, etc.

Shortcut:

- 62	6	⊢-	
7		۰.	à.
- 1			1
- 1			- 1
	<u> </u>		

Click on the toolbar.

For all curves, choose "<u>0</u> - none" for the line type, and select the symbol type you want to use for each curve. Alternatively, if this is an operation you will be performing frequently, you can automatically turn off all lines and set symbol types with a <u>macro</u>. The following macro turns off line segments for **all** curves and sets symbol types for 6 curves to the first six symbol types.

LineType(-1,0) SymbolType(1,1) SymbolType(2,2) SymbolType(3,3) SymbolType(4,4) SymbolType(5,5) SymbolType(6,6)

#### 2.6.9 How do I replace the numbers along the axes with custom text?

Internally all DPlot X and Y values are numbers only. Even in the case of calendar dates, the dates are represented internally with a numeric code. But you can replace the numbers appearing along the X and/or Y axes with labels of your own choosing by using the *X*, *Y Labels* command on the *Text* menu.

For example, select the table below with your mouse, right-click and select Copy.

Average Sales Per Day 108.42 279.39 546.27 375.3 137.61

In an empty XY graph document in DPlot, select *Edit>Paste*. This will result in a line graph. Rightclick on the graph and select *Bar Chart*. Right-click again on the graph and select *Bar Chart Options* and use the settings shown below.

Bar Chart Options		X
Stacked bars         ✓ Amplitude labels         ✓ Percentage labels         ✓ Labels outside bars         Bars span intervals in ≚ (1 data set)         ✓ Unique colors for each bar (1 data set)         Base of bars at minimum Y extent         No black border on filled bars         Shaded bars         Bar width:         100         ▲ roup w	ata sets et only) a set only) width: 100 📩	OK Cancel Help Make defaults
for <u>D</u> ata set: Curve #1	Pattern ← Hollow ← Solid ← Horizontal lines ← Vertical lines ← 45-degree down ← 45-degree up ← H-V cross-hatch ← 45-degree cross	

Other than color settings (which you can change with <u>Options>Colors>Edit</u> or by clicking on the Colors button), your graph should now resemble the following:



In our case each entry represents a unique type of fruit, rather than any numeric value. To place labels along the X axis below each bar showing the fruit type for each bar, select  $\underline{\text{Text}>X, Y}$  Labels and make the following entries:

X Labels			×	
You can enter text labels to use in place of or in addition to the numbers, dates, or times that appear on the X or Y axes. These labels will be drawn with the same orientation as numbers along the corresponding axis (controlled w/ Number Formats command). The X/Y value entry refers to the primary X/Y data values only, regardless of whether multiple axes				
are used. To use these labels ONLY, set the number format for the corresponding axis to "None" using the Number Formats command.				
X∕Y value	Label	Lo <u>c</u> ation	<u>O</u> ffset, inches	
0.5	Apples	X bottom	0	
1.5	Bananas	X bottom	0	
2.5	Oranges	X bottom	0	
3.5	Grapes	X bottom	0	
4.5	Limes	X bottom	0	
		<b>•</b>		
		-		
		-		
		-		
		-	<b>•</b>	

The labels will be centered at the X (or Y) values entered. That's why we've used 0.5, 1.5, etc.: those are the midpoints of each bar with the bars filling the space between X values.

Finally, to eliminate the numeric entries along the X axis, right-click any number on the X axis and select *None*. The finished product:



## 2.6.10 How do I set error bars with unique error values?

Three of the four error bar error amount options (see <u>Error Bars</u> command on the Options menu) use some fixed error value or a proportion of the curve's amplitudes. The fourth option, though, allows unique error values at every data point. To make use of this feature first create a new curve whose amplitudes (Y values) are set to the error values. If you want both positive and negative error values and those values should be different, create two new curves: one with the positive error values and the other with the negative error values. The X values of these "curves" are irrelevant: the n'th point in the error source curve(s) will be used as the error value for the n'th point in the selected curve. The source curve (s) for the error values will not be drawn, nor will they be considered when finding a plot's extents. In most dialog boxes that show lists of curves, these error source curves will then generally be shown with a suffix of "(errors)".

Both the positive and negative error values should be positive.

#### Example:

Select the following table with your mouse, right-click and select Copy, then Paste into DPlot.

Х	The data	+ Errors	- Errors
0.000	0.000	0.0342	0.0910
0.167	0.500	0.0619	0.0711
0.333	0.866	0.1060	0.0762
0.500	1.000	0.1133	0.0352
0.667	0.866	0.0472	0.1056
0.833	0.500	0.0666	0.0037
1.000	0.000	0.0426	0.0649

You should end up with a plot resembling the following (extents adjusted for clarity):



Select Error Bars on the Options menu:

🕅 Error Bars		X
Apply to all curves Which curve?  T: The data C: + Errors 3: - Errors	Vertical Horizontal	OK Cancel Help
Bar <u>w</u> idth: 0.25'' Pen width: 0.03''	Error Amount Fixed value Percentage Custom U U U U U U U U U U U U U U U U U U U	
	Different error values for every point:     Positive errors from: 2: + Errors Negative errors from: 3: - Errors	

Key points:

• Select the first curve (in this example, or whatever curve you want error bars for) under "Which curve?"

- Select "Both" under "Display" if you want both positive and negative error bars. If you want only positive errors, select "Plus", and likewise if you want only negative errors, select "Minus".
- Under "Error Amount" select "Different error values..." and choose the appropriate curves for the positive and negative error values. In our case we've selected the 2nd curve (the 3rd column of values in our original data above) for the positive error values and the 3rd curve for the negative errors.

• Change the "Bar width" and "Pen width" entries to suit your needs, then click OK.

The result:



If instead of (or in addition to) vertical bars you want horizontal error bars, click the Horizontal tab. You can have both vertical and horizontal error bars on the same curve. The error amplitudes work in the same way: Y values from the error source curves are used for the error values.

# 2.7 Scaling

#### 2.7.1 How do I create a polar plot?

If you want, for example, a polar plot of the equation Y=abs(sin(3*X)) similar to this:



... you'll likely find that when your data is first imported or you use Y=f(X) to generate your data, you

get a plot with linear X, linear Y scaling similar to this instead:



Right-click on the graph and select *Polar Coordinates*, or select *Options>Linear/Log Scaling>Polar Coordinates*. When using the Y=f(X) or similar commands, DPlot will automatically correctly set the scale for the angular values to either degrees or radians. If the source of your data is instead a text file or data pasted from the clipboard, then the angular values may be in the wrong units (radians are the default when DPlot is first installed). If the wrong units are assumed, the problem is usually obvious:



This is easily fixed. Right-click on the plot. At the bottom of the dropdown menu, select <u>Polar Plot</u> <u>Options</u>. (This option is also available on the main menu under *Options>Linear/Log Scaling*). Check *degrees* under *Angular Units*. Other formatting options specific to polar plots are also available on this dialog box.

Note that whether the angle input is in degrees or radians, the angular axis labels are in degrees. A handy option for normal linear scales with trigonometric functions or, in our case, polar coordinates is to instead display the angular labels using either multiples of PI or fractional values of PI. Select the <u>Number Formats</u> command on the Options menu, ensure X axis is checked, and in the Format list select "Pi Fractions".

Take a few moments to experiment with different settings in this dialog. For more information on the *Number Formats* options, click <u>here</u>.

## 2.7.2 How do I create a bar chart?

A bar chart or bar graph is an XY plot with rectangular bars of lengths proportional to the value they represent. When you first create an XY plot the graph will always be initially drawn as a line graph with line segments between each data point. To switch to a bar chart, right-click within the box surrounding the plot and select "Bar Chart".

#### Example:

Select the data below, right-click and select Copy, then Paste into DPlot.

Floor-Load Data		
Live load, psf	Test 111902-01	Test 111902-02
0	10	13
20	17	15
40	41	34
60	42	37
80	35	40
100	19	26

120	22	20	
140	16	18	
160	б	8	
180	7	б	
200	2	1	
220	3	2	

How do I ...?

99

As mentioned above, you will initially see a line graph with line segments between each data point:



## Floor-Load Data

Right-click within the box surrounding the plot and select "Bar Chart":



Alternatively, select Linear/Log Scaling>Bar Chart on the Options menu. In either case this action will result in a bar chart with default settings:



Floor-Load Data

To change bar chart-specific settings right-click within the plot again and select Bar Chart Options. Each of the options are thoroughly described in the <u>Bar Chart Options</u> topic. Several variations:



Floor-Load Data Stacked bars





Each bar (or group of bars) will be drawn centered between the associated X and that X plus the 100% bar width. If you want the bars to begin at the associated X value, you can modify X to accomplish this. Use Edit>Operate on X with X=X-(\$X(1,2)-\$X(1,1))/2 to shift the bars left by 1/2 the distance in X between the first 2 points in the first data set.

## 2.7.3 How do I create a grain size distribution plot?

A grain size distribution plot is an XY plot with, typically, a reversed logarithmic scale on the X axis showing grain sizes and a linear percent finer scale on the Y axis. In addition, DPlot shows material classifications below the X axis and sieve sizes above the graph. An example of this type of plot is shown in example plot ex07.grf:



#### Example:

Select the data below, right-click and select Copy, then Paste into DPlot.

Grain Size, mm	Percent	Finer
9.49431	100	
6.3489	90.0497	
4.76415	79.5579	
3.25503	65.4762	
1.95326	48.5495	
1.19096	37.2581	
0.823269	31.4927	
0.580235	27.3879	
0.419289	24.2034	
0.302874	20.4645	
0.208457	17.2875	
0.154445	15.0234	
0.105706	12.5868	
0.0741779	11.0704	

You will initially see a normal linear X, linear Y line graph:



Right-click within the box surrounding the plot and select "Grain Size Distribution":



Alternatively, select Linear/Log Scaling>Grain Size Distribution on the Options menu. In either case this action will result in a grain size distribution plot with default settings:



To change type-specific settings right-click within the plot again and select Grain Size Options. Each of the options are thoroughly described in the <u>Grain Size Options</u> topic. There you can choose to display Udden-Wentworth scale material classifications rather than the default AASHTO-standard sizes:



and/or British standard sieve sizes at the top of the plot instead of the default US standards:



You can also elect to display a custom list of sieve sizes at the top of the plot, or to draw bold vertical lines across the plot at material boundaries.

# 2.7.4 How do I specify a tick mark interval?

To set a specific tick mark interval, use the Extents/Intervals/Size command on the Options menu and:

1) Select Specify Interval

OR

2) Select **Specify number** and enter the number of intervals on each axis. Check **Specify extents** and enter the extents for the X and Y axes such that the range in magnitudes divided by the number of intervals is equal to the desired tick mark interval.

The resulting tick mark interval may not be exactly as described above. DPlot will adjust the extents set with the Extents/Intervals/Size command such that the tick mark interval is some multiple of 1.0, 1.5, 2.0, 2.5, 3.0, 4.0, 5.0, 6.0, or 8.0 times a power of 10.

For example, if 6 tick mark intervals are specified on an axis and the extents are set to 0.0 to 10.0, DPlot adjusts the upper extent to 12.0 so that the tick mark interval is 2.0 rather than 1.6667.

# 2.7.5 How do I force scaling on the X axis to be equal to the scaling the Y axis?

In other words, how do you set scale factors such that the physical distance for a given range

between values on the X axis is the same as the physical distance for the same range in values on the Y axis?

For XY plots, DPlot initially uses as much of the display as possible for a plot. For this reason, the physical distance between equal values on the X and Y axes will generally not be the same. You can force a 1:1 scaling by using the <u>Extents/Intervals/Size</u> command on the *Options* menu.

#### For linear X and Y scales

Check both the **Specify extents** and **Specify plot size** boxes. Enter the desired extents on both the X and Y axes under **Specify extents**, then use proportional values for the plot width and height under **Specify plot size**. For example, if the X axis extends from -2 to 8 (10 units) and the Y axis extends from 4 to 8 (4 units), then the plot width should be 10/4 = 2.5 times the plot height. Conversely, if you want the plot width and height to be specific values (square, for example), then change the plot extents so that they are proportional to the width and height. For example, if the plot width and height are set to 6 and 4 inches, respectively, then the extents on the X axis should be 6/4=1.5 times the extents on the Y axis.

You can also have DPlot automatically adjust the plot extents for linear scales by, again, checking both the **Specify extents** and **Specify plot size** boxes, then clicking the **Set X:Y=1:1** button. DPlot will decrease either the width or height of the plot to get the desired aspect ratio. Note that this setting is **not** persistent; if you subsequently change the extents or the size values or zoom in/out on a plot, the 1:1 relationship will not necessarily be maintained.

#### For logarithmic X and Y scales

If you do not care about the actual physical dimensions of the X and Y axes and want to allow DPlot to use the full extents of the display and/or printed page, simply check the *Square Log Cycles* item on the *Options* menu. You'll also need to ensure that **Specify plot size** on the *Extents/Intervals/Size* dialog is **unchecked**.

If you would rather specify the plot size, then the steps to take are similar to those for linear scales except that you will be comparing the logarithms of the axis extents rather than the displayed numbers. For example, if the specified plot extents are (1,1) to (1000,100), then the plot width should be  $(\log(1000)-\log(1))/(\log(100)-\log(1)) = 1.5$  times the plot height. Conversely, if you want the plot width equal to the plot height, you can adjust the extents accordingly. In the previous example you could change the Y axis extents to 0.5 to 500 (or 1 to 1000 or 0.1 to 100, etc.) so that the range in logarithms is identical to the range in logarithms on the X axis.

#### For Mercator projections

DPlot will **force** the size of the plot to get the proper relationship between physical distances on the X and Y axes, regardless of whether you've used the <u>Extents/Intervals/Size</u> command to set the extents and/or size. But the changes it makes will not necessarily be desirable: the plot might be too wide or too high to suit your needs. For the best, most predictable results, as with linear and logarithmic scales use the <u>Extents/Intervals/Size</u> command, check both the **Specify extents** and **Specify plot size** boxes and enter the associated values, then click the **Set X:Y=1:1** button.

## 2.8 Startup Options

#### 2.8.1 How do I specify options on the DPlot command line?

To modify how DPlot starts on a one-time basis, you can add switches to the Windows Run command (Start menu). If you plan to use a modified startup method frequently, you can create a shortcut on the Windows desktop.

#### To add switches to the Windows Run command:

- 1) On the Windows Start menu, click Run.
- 2) Enter the path to DPlot, such as C:\Program Files\DPlot\DPlot.exe, or click Browse to locate it.
- 3) At the end of the path, type a space, and then type a startup switch. (see below)
# To create a desktop shortcut:

- 1) Right-click the Windows desktop, point to New, and then click Shortcut.
- 2) In the *Type the location of the item* box, enter the path to DPlot, such as C:\Program Files\DPlot\DPlot.exe, or click Browse to locate it.
- 3) At the end of the path, type a space, and then type a startup switch.

# Startup switches

<u>Switch</u>	Description	
<b>If</b> filetype filename	Opens and displays the file <i>filename</i> of type <i>filetype</i> (see below)	
<b>/p</b> filetype filename	Prints <i>filename</i> using the current default printer and printer options. <i>Filetype</i> is described below.	
<i>Im macroname</i>	Runs the macro named macroname (name as it appears on the Macro dialog)	
<b>Is</b> preferences file or <b>Isp</b> prefences file	Reads preferences from "preferences file", which was previously saved using the Save Preferences command on the File menu. If the form <b>/sp</b> is used, this preferences file will be used on <b>all</b> new documents during the current DPlot session. %APPDATA%\DPlot\Preferences\default.set will in this case be ignored. Without the <b>p</b> suffix, the preferences file will only be used on documents opened on the command line.	
<b>/d</b> folder	Specify the default folder for opening and saving files.	
	This switch is primarily useful for running macros from the command line when you do not want to manually edit the macro to include a <u>Directory</u> command. For example, if you have a macro named SavePNG that contains:	
	<pre>ForFilesIn("*.grf")    FileSaveAs("Portable Network Graphics",".png")    FileClose() NextFile</pre>	
	then the command line:	
	[d:\path\]dplot.exe /d "c:\My Data Folder" /m SavePNG	
	will open all files with extension .grf in the folder c:\My Data Folder and save PNG images of those graphs to c:\My Data Folder.	
/nonag	Prevents the introductory dialog from being displayed (tip of the day and/or expiration days for the trial version). This might be useful if you are the author of an application that communicates with DPlot via dynamic data exchange, since those dialogs interfere with DDE. (DPLOTLIB.DLL automatically applies /nonag when starting DPlot)	
<b>/show</b> state	Overrides the default show state of the DPlot window. <i>state</i> =0 to hide DPlot, 2 to start minimized, 3 to start maximized, or 9 to start "restored" (not maximized). For programmers, these are the same values as the <i>nCmdShow</i> argument to the ShowWindow function.	
<b>/xcy</b> c,c,c,c	Specifies how columns in a multiple-column ASCII text file (type 4 below) are interpreted. Use 0 following the x if X values should be generated by DPlot, starting at 0 and incremented by 1 for each data point. Y column indices should be separated by commas. No spaces are allowed. For example, <b>/x2y3,4,5</b> will use the 2nd column in the file for the X values and the 3rd, 4th, and 5th columns for Y in 3 curves. This setting <b>only</b> affects command line file operations; it is not persistent. This setting should precede any command line filenames.	

*Ixcyczc* Similar to above, but applied to 3D surface plots (type 11 below).

*filename* Same as the *I***f** switch above, but forces DPlot to attempt to determine the filetype. See below.

# NOTE: As with any other Windows program, if a filename and/or path contains spaces then it should be surrounded by double quotation marks, as in:

dplot.exe "c:\my documents\my data\datafile.csv"

File type codes

- 1 DPlot file
- 2 ASCII file, one data set with arbitrarily spaced points.
- 3 ASCII file, one data set with evenly spaced points.
- 4 ASCII file, multiple Y values for each X.
- 5 Unformatted single-precision (4 byte floating point) data.
- 6 Binary file produced by Pacific Data Model 9820 recorders, DNA/Bendix format.
- 7 Binary file produced by Pacific Data recorders, **OLD** format.
- 8 Binary file produced by Pacific Data recorders, **NEW** format.
- 9 Nicolet Time Domain files (binary) created with the Nicolet System 400 Digital Oscilloscope.
- 10 Hardened Data Acquisition System (HDAS) files.
- 11 Random 3D values (contour plot).
- 13 1D statistics data.
- 267 Random 3D values. Identical to 11, but interpreted as a 3D scatter plot rather than a surface plot.

File types are described in more detail in the description of the Open command command.

If /F filetype is omitted, DPlot attempts to determine the file type with the following assumptions:

If The first line of the file contains the DPlot file header (DPlot $vxx$ )	DPlot assumes the file is: DPlot file (type 1)
Plugin_typecheck for any installed file import <u>plugin</u> returns true.	type 12 (plugin). plugin_readfile in the corresponding plugin DLL is used to read the file.
ASCII file with one or more columns of data	type 4
<ul> <li>a) The file size is a multiple of 1024 bytes</li> <li>b) The file size is at least 2048 bytes) The first four bytes of the file has the number of data points, and the file size is equal to 1024*(int((NumPoints- 1)/128)+2)</li> </ul>	type 5

If the file does not match any of these formats, DPlot attempts to identify it as one of the five supported data recorder binary file formats using a combination of expected file size and file header information. Note that if the file type is not specified, DPlot will always identify type 2 and 3 files (see below) as type 4 files. Type 2 files are a subset of type 4, so the data will be interpreted correctly. However, type 3 files that are identified as type 4 will lack the spacing between points (X will start at 0 and be incremented by 1 for each data point).

See also: DPlot File

# 2.9 Miscellaneous

# 2.9.1 How do I pan from side-to-side (or up-down) on a plot?

Very easily! Use your keyboard's arrow keys to pan the plot 1/2 width or height at a time. Press the CTRL key in conjunction with the arrow keys to pan 1/20th of the extents. This feature works **only** on linear or logarithmic scales of XY plots, Mercator projections, and 2D views of surface plots.

If <u>Click and drag to zoom</u> on the General Options screen is unchecked, you may also pan by clicking and dragging the mouse. When using the mouse, you can restrict the panning motion to horizontal, vertical, or 45 degree increments by simultaneously pressing the Shift key. You can forego use of the Shift key to restrict the panning direction by checking the **Restrict mouse panning to H/V/45°** option on the <u>View menu</u>.

**NOTE:** Logarithmic scales will always pan an integer number of log cycles, regardless of the extents of the plot.

# 2.9.2 How do I temporarily hide one or more curves on a crowded document?

On the View menu, select "Hide curves...", then select the curves that you want to hide. All unchecked curves will remain visible. Visibility may be restored with the "Unhide all curves" command on the View menu.

Alternatively, if your document has a legend you may right-click on the legend entry for the curve you want to hide, then select "Hide <label>", or to hide all curves **other** than the selected one, select "Hide all but <label>" from the drop-down menu:



And as another alternative, you can right-click on a curve and select "Hide" from the drop-down menu:



You can also hide a curve by clicking the middle mouse button while the mouse points to a curve, hide all curves **except** the one pointed to by pressing CTRL + the middle mouse button, and make all curves visible by clicking the middle mouse button while the mouse does **not** point to a curve.

# 2.9.3 How do I distribute my DPlot files to others?

You are of course welcome to distribute any files you produce using DPlot to others. If you want users to be able to view your DPlot .GRF files (as opposed to saved bitmaps, metafiles, etc.) then those users need a viewer. The solution is DPlot Viewer.

DPlot Viewer is a free scaled-down version of DPlot intended for use by developers that want to distribute DPlot graphs to others. DPlot Viewer will display and print any graph produced by DPlot or DPlot Jr. DPlot Viewer may be redistributed to end users without royalty payments or other restrictions.

For more information and/or to download DPlot viewer visit the DPlot web site.

# 2.9.4 How do I get help within DPlot?

Context-sensitive help is available for most DPlot menu commands and selected regions of the screen. Following is a description of the help-related keys:

#### Keys(s) Function

- F1 Gets Help and displays the Help Index for DPlot. If the Help window is already open and active, pressing F1 displays the "How to Use Help" topics. Pressing F1 while a menu command is highlighted displays a Help topic on the selected command.
- SHIFT+F1 Changes the pointer to So you can get Help on a specific command or screen region. You can then choose a command or click the screen region you want to know more about.

Escape Changes the pointer to the normal  $\Bbbk$ .

The Help button on the toolbar toggles context-sensitive help.



# 3 License and Ordering Info

Evaluation License Copyright/License/Warranty DPlot Ordering Information DPlot Order Form Register DPlot

# 3.1 Evaluation License

DPlot® Copyright © 2001-2014 by HydeSoft Computing, Inc. All rights reserved.

This license applies to the evaluation version of DPlot. If you have purchased a license for the registered version of DPlot, see the section <u>Copyright/License/Warranty</u>.

# **License Agreement**

Carefully read the following terms and conditions before using this software. Unless you have a different license agreement signed by HydeSoft Computing, Inc. your use of this software indicates your acceptance of this license agreement and warranty.

# **Evaluation and Registration**

This is not free software. Subject to the terms below, you are hereby licensed to use this software for evaluation purposes without charge for a period of 30 days. If you use this software after the 30 day evaluation period you must purchase a license. Payments must be in US dollars drawn on a US bank, and should be sent to HydeSoft Computing LLC, 110 Roseland Drive, Vicksburg, MS 39180. Credit card ordering and quantity discounts are available, as described in the section <u>Ordering Information</u>. When payment is received you will be sent a registration code which will unlock additional features and remove the startup registration notice.

Unregistered use of DPlot after the 30-day evaluation period is in violation of U.S. and international copyright laws.

# **Disclaimer of Warranty**

This software and the accompanying files are offered "as is" and without warranties as to performance or merchantability or any other warranties whether expressed or implied. Because of the various hardware and software environments into which DPlot may be put, no warranty of fitness for a particular purpose is offered. Warranties of merchantability or of fitness for any purpose are specifically disclaimed.

Good data processing procedure dictates that any program be thoroughly tested with non-critical data before relying on it. The user must assume the entire risk of using the program. Any liability of the seller will be limited exclusively to product replacement or refund of purchase price.

# **Customer Remedies:**

HydeSoft Computing's entire liability and your exclusive remedy shall not exceed the price paid for the licence to use the product or one United States Dollar, whichever is the greater.

# **Distribution**

You are hereby licensed to make as many copies of this software and documentation as you wish; give exact copies of the original software to anyone; and distribute the software and documentation in its unmodified form via electronic means. There is no charge for any of the above.

# **Prohibitions**

You are specifically prohibited from:

- Distributing registration codes for this product or "keygen" software designed to circumvent the registration process.
- Publishing registration numbers and/or security keys on a publicly-accessible web site. Though the
  intent may be to distribute this information to legitimate site-license users only, this practice is strictly
  prohibited.
- Charging or requesting donations for any copies, however made.
- Distributing the software and/or documentation in whole or in part with other products (commercial or otherwise) without prior written permission.

# 3.2 Copyright, License, & Warranty

DPlot® Copyright © 2001-2014 by HydeSoft Computing, Inc. All rights reserved.

This End User License Agreement (EULA) applies to the registered version of DPlot. If you are using an evaluation version of DPlot, see the section titled <u>Evaluation License</u>.

Use of DPlot Graph Software is subject to the following terms and conditions. If you do not accept these terms, you must cease use of the Software. Continued use of the Software indicates acceptance of these terms.

# **Ownership of the Software**

 DPlot Graph Software ("the Software") and the accompanying files are owned by HydeSoft Computing LLC and are protected by United States copyright laws, by laws of other nations, and by international treaties.

# **Limited Warranty**

- 2. The Software is provided "as is". HydeSoft Computing disclaims all warranties, either express or implied, including but not limited to implied warranties of merchantability, fitness for a particular purpose, and non-infringement. The entire risk for the use of the Software is assumed by the user.
- 3. HydeSoft Computing does not assume liability for damages, including loss of profit, lost savings, damage to equipment or data, or other incidental or consequential damages arising out of the use or inability to use the Software.
- 4. HydeSoft Computing's entire liability and your exclusive remedy shall be return of the price paid for the Software.
- 5. This Limited Warranty is void if failure of the Software has resulted from modification, accident, abuse, or misapplication.

# **Restrictions on Use**

- 6. HydeSoft Computing grants the right for a single user to use the Software. You may install the Software on more than one computer provided that only the licensed user is using the Software. Network installation or distribution within an organization is only permitted if you have purchased licenses sufficient to cover the number of copies in use, whether simultaneously or not. You may make backup copies of the Software for your personal use only.
- 7. You are authorized to distribute copies of the Software to others within your organization who are licensed by the same DPlot license. All other distribution of the Software in whole or in part is strictly prohibited. **Exception**: Organizations that have prior written authorization from HydeSoft Computing to distribute DPlot with their own software and/or products are entitled to distribute the software and documentation in its unmodified form.
- 8. You may not reverse engineer, decompile, or disassemble the Software. You may not modify the Software in any manner.

- 9. You may not distribute registration codes for the Software to users not covered by your purchase or "keygen" software designed to circumvent the registration process.
- 10. You may not publish registration numbers and/or download locations on a publicly-accessible web site. Though the intent may be to distribute this information to legitimate site-license users only, this practice is strictly prohibited. Distribution of this information to authorized users via e-mail or a private password-protected web site is permissible.
- 11. You may not copy the Software, except that you may make a copy of the Software solely for backup or archival purposes.
- 12. The trial version of DPlot Graph Software may be freely distributed under the conditions that it is not modified in any way and that no remuneration, monetary or otherwise, is received for the trial version. For more information concerning distribution of the trial version of the Software, see <u>Evaluation License</u>.

# **Other Rights**

- 13. Any suggestions concerning the Software which you send to HydeSoft Computing become the property of HydeSoft Computing.
- 14. HydeSoft Computing provides free technical support by email to DPlot users. However, we reserve the right to terminate support to users with unusual system problems beyond the scope of the Software.
- 15. HydeSoft Computing reserves all rights not specifically granted in this license.
- 16. This Agreement is governed by the laws of the State of Mississippi, United States of America.
- 17. Notice to U.S. Government End Users. The Software and Documentation are "Commercial Items," as that term is defined at 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. HydeSoft Computing, 110 Roseland Drive, Vicksburg MS, USA.

# 3.3 DPlot Ordering Information

**Please note:** Registration numbers and download instructions for the licensed version of DPlot will always be sent via e-mail, regardless of the method of payment or ordering. Therefore it is essential that a valid e-mail address be provided with your order. Customer information is considered confidential and will never be shared or distributed to any third party.

All orders are subject to the DPlot License Agreement.

#### Immediate Online Delivery

Immediate online delivery of a registration number and download instructions for the licensed version of DPlot are available for credit card orders, wire transfers, and other payment options placed via the Internet using the **Buy Now** link on the DPlot home page: <u>http://www.dplot.com/registration.htm</u>. If you do not want to or are prohibited from entering credit card information online, our payment processor also accepts credit card orders by **FAX** or **phone** or **PayPal** orders, all for a small processing fee. **Wire transfers** are also available. To select a method of payment other than the default "Secure Online Credit Card", select an option from the "Method of payment" box on the order page:

Phone* :		Ext:
Method of payment :	Secure Online Credit Card 🚽	
Please fill the form below i shipping address is differe	Secure Online Credit Card Wire Transfer Phone Fax Mail / Check / Money Order PayPal Local Bank Transfer	or if the

# **Ordering by Check**

To order by check, send the <u>order form</u> and a check payable to HydeSoft Computing LLC, 110 Roseland Drive, Vicksburg, MS 39180 USA. Payments must be in US dollars drawn on a US bank, or you can send international postal money orders in US dollars. Click <u>here</u> to view a ready-to-print Order Form.

# **Site licenses**

A site license for DPlot entitles an organization to receive one registration code and duplicate this code as necessary for use within the organization on the specified number of computers. See the table below for site license pricing.

# Pricing

**NOTE:** Prices are subject to change. Visit the <u>Buy Now</u> page on the DPlot web site for current pricing information.

<u>DPlot Jr</u>	free		
DPlot Viewer	free		
DF	Plot Full Versio	n	
Number of users	Download only	CD delivered by USPS	
1	\$195	\$205	
2 to 9	\$150/user	\$159/user	
10 to 24	\$120/user	\$108/user	
25 to 49	\$106/user	\$113/user	
50 to 99	\$97/user	\$103/user	
100 to 249	\$87/user	\$92/user	
250+	\$75/user	\$79/user	
Jr. High and High Schools	\$195/school	\$205/school	
College/University Subscriptions			
1 year	\$300		

\$600

3 years

# 3.4 DPlot Order Form

To order by check, print, fill out and send this order form and a check payable to:

HydeSoft Computing, LLC 110 Roseland Drive Vicksburg, MS 39180 USA

Payments must be in US dollars drawn on a US bank, or you can send international postal money orders in US dollars. All orders are subject to the <u>DPlot License Agreement</u>. Prices are subject to change; for the latest price information visit the <u>DPlot web site</u>. Customer information is considered confidential and will not be shared or distributed to any third party.

Item	Users/ Quantity	Unit Price download only	CD delivered by USPS	Total
1 user	1	\$195 =	add \$10	
2 to 9 users		@\$150 per user =	add \$9 per user	
10 to 24 users		@\$120 per user =	add \$8 per user	
25 to 49 users		@\$106 per user =	add \$7 per user	
50 to 99 users		@\$97 per user =	add \$6 per user	
100 to 249 users		@\$87 per user =	add \$5 per user	
250+ users		@\$75 per user =	add \$4 per user	
Jr. High/High School	N/A	\$195 =	add \$10 (1 CD)	
1 year College/University Subscription	N/A	\$300 =	add \$10 (1 CD)	
3 years College/University Subscription	N/A	\$600 =	add \$10 (1 CD)	
Total =				
Discounted upgrade price. If upgrading from an older version, subtract 2/3				

Name:	
Company/School:	
Mailing address:	
City, State, Zip:	
Country:	
E-Mail address:	
How did you hear about DPlot?	

# 3.5 Register DPlot

## If you paid the DPlot registration fee:

If you paid the DPlot registration fee and received a registration number from HydeSoft Computing or an authorized reseller, OR if you received a free copy of DPlot from an authorized distributor, please enter your name and registration number **exactly** as they appeared on the instructions you received with DPlot. **You cannot substitute a different spelling for the name or use a different name.** If entered correctly, the registrant's name will appear in the About dialog box, and the registration reminder dialog box will no longer be displayed when you start DPlot. Please save this information for possible use with future upgrades, or in case you re-install DPlot.

Note: If your name is spelled incorrectly in the instructions, please write to the address in the About dialog box with the correct spelling, and you will receive a new registration number.

### If you have not yet paid the DPlot registration fee:

If you downloaded an evaluation version of DPlot, or if you received this version of DPlot on a disk or CD with a book or with other hardware or software, and you have not paid the DPlot registration fee, you are licensed to use DPlot for evaluation purposes only, subject to the Evaluation License.

For information on purchasing a license to use DPlot for non-evaluation purposes, see the section Ordering Information.



# 4 Menu Commands

#### To get information about individual menu commands

- 1. On the toolbar, click the Help button.
  - Help button

The mouse pointer changes to a question mark:

Relp pointer

2. Click the menu you want, and then click the command name for which you want information. DPlot displays Help for that menu command.

## See also:

File Menu Commands Edit Menu Commands Text Menu Commands Options Menu Commands Generate Menu Commands Info Menu Commands View Menu Commands Window Menu Commands

# 4.1 File Menu

#### <u>New</u>

Opens a new document window.

### <u>Open...</u>

Opens a new document window; reads data from a file

## Append...

Adds data from file to existing document window

<u>Close</u>

Closes the currently active document window.

# Save

Saves the active document's data in a DPlot file.

# Save As...

Saves the active document's data.

### **Get Preferences...**

Retrieves plot settings from a previously saved file.

# Save Preferences...

Saves plot settings (line and symbol styles and sizes, scaling, etc.) to a file.

#### Print...

Displays a Print dialog box; prints the current plot

#### Page Setup...

Changes margins, paper size, and orientation that will be used when printing a plot.

# Related macro commands

**FilePageSetup** 

# Macro...

Build and/or run a macro file to perform unattended operations.

### **Resize Arrays...**

Changes the maximum number of curves and points per curve.

<u>Exit</u>

Quits DPlot

# 4.1.1 New command

Opens a new document window. If you have previously saved document preferences to a settings file named DEFAULT.SET in the Preferences folder below the Application Data\DPlot folder (using the Save Preferences command on the File menu), the new document window will use those attributes.

For XY plots DPlot attempts to allocate arrays for the X,Y data pairs using the maximum number of curves and points last entered using the <u>Resize Arrays</u> command. If there is not sufficient memory for allocating those arrays, DPlot presents an error message explaining the problem. This might happen if you have many plots each containing many millions of data points. You may correct the problem by:

- 1) Entering smaller values for either or both of the number of curves and number of points, then selecting OK.
- 2) Select Cancel, close one or more document windows, and select New again.

# Shortcut:



<u>FileNew</u> <u>FileNewEx</u>

# 4.1.2 Open command

Allows you to open a DPlot file (containing data, strings, and formatting information), or one or more of nine other file formats.

#### Shortcut:



- A) <u>DPlot file</u> Contains data for one or more curves, as well as the title, axis labels, legend, and other formatting information. You may also select this option to open a compressed DPlot file.
- B) ASCII file, one or more data sets (X,Y curves) with arbitrarily spaced points. The first line should have the number of points (N). Subsequent lines should contain each of the N X,Y data pairs. Data values may be separated by one or more spaces or a comma.

Example:



You can include multiple curves by repeating the pattern: The last X,Y pair for curve 1 is followed by N for curve 2, etc.

#### Example:

5

1.00,1.00 2.00,4.00	1000
3.00,9.00	
4.00,16.0	
5.00,25.0	
5	100
1.00,1.00	
2.00,8.00	
3.00,27.0	
4.00,64.0	10
5.00,125.	
5	
1.00,1.00	
2.00,16.0	1.0
3.00,81.0	1 2 3 4 5 6
4.00,256.	
5.00,625.	

C) Constant DX. The number of points (N) and spacing between data points (DX) should be on the same line, or DX should be on the line following N. DPlot will allow up to 3 header lines before the line containing N. If found, those lines will be used as the plot's title lines. The Y values may be on the same line separated by one or more spaces or a comma, or on separate lines. The first X value is assumed to be 0. (This can be amended by selecting 'Add a constant to X' from the 'Edit' menu.)

# Example:



D) Multiple column text files.

Multiple column text files (type D on the **Select File Type** dialog) may have columns delimited by commas (CSV files, for example), spaces, tabs, or semicolons. If the <u>decimal symbol</u> is set to a comma, columns should be separated by tabs or semicolons. Files may contain up to 30 header lines preceding the data columns (maximum number of header lines can be increased with the <u>General</u> command on the Options menu). DPlot looks for up to 20 successive lines that have the same number of data columns and the same data type in each column, and, if found, decides that this must be where the data starts. Data values may be separated by one or more spaces, a comma, or a tab. DPlot will read up to a maximum of 100 columns of data, restricted to 8192 characters per line.

**Column Headings.** If the data starts after line 1, DPlot attempts to get column headings from the previous line. Column headings are used as the legend for multiple curves, or the X and Y axis labels for a single curve. For this feature to work as expected with headings containing spaces, the columns must either be comma-separated, tab-separated, or delineated with double quotation marks. Otherwise DPlot will assume the labels are delineated with spaces.

**Title Lines.** Up to the first 3 lines in the file are used as title lines for the plot, unless these lines consist of numbers and/or spaces only.

**Column Interpretation.** If the file consists of a single column of data DPlot interprets the data as Y values, setting the corresponding X values to the index of the Y value in the file, starting from 0. By default, if the file consists of more than one column of data, DPlot uses the first column for the X array and subsequent columns as separate Y arrays. For multiple-column files, and/or for files consisting of alternating X,Y columns, you can change this default behavior by checking the **Pick Columns to Plot** box (see below) on the Open dialog box.

Example:



Example (column headings):



■ Example (column headings, comma-separated):



In addition to numbers DPlot will also accept columns consisting of dates, times, date-time pairs, currencies, and percentages. Date format is flexible but entries must be separated by a dash (-) or

© 2014 HydeSoft Computing, LLC

forward slash (/). If the month is specified as a number, then DPlot assumes the order is m/d/y (or d/m/y if *Assume input dates are of the form d/m/y* under the <u>General</u> command on the Options menu is checked) unless you use 4 digit years or if the assumed month entry is greater than 12. These date forms (for Jun 8, 2005) are acceptable:

6/8/05 06/08/2005 6-8-2005 2005/6/8 8-Jun-05 June-08-2005

but 'June 8, 2005' is not.

2-digit years less than 90 are interpreted as 21st century dates; 2-digit years greater than or equal to 90 are interpreted as 20th century dates. Of course, to avoid any ambiguity a 4-digit year is preferable.

Time values should be in the form h:m:s AM/PM, using ':' as the separator. Leading zeroes are acceptable. If the AM or PM designation is omitted, DPlot assumes a 24 hour clock.

Example (time values):



Date-time pairs should be in the form 'm/d/yy h:m:s AM/PM'.

Only dollar signs and British pound signs are currently accepted as currency symbols. The currency symbol should precede the value, as in '\$56.23'. If monetary values include a comma for the thousands separator, the entry **must** be surrounded by double quotation marks. This is the same scheme used by Excel when saving CSV files.

Percentages should be followed by a percent sign (%). DPlot divides the number by 100 and uses Percent <u>number formatting</u> on the associated axis when appropriate.

Any other data type will be ignored, but in general will be allowed. If present, blank entries are ignored. (For this feature to work as expected, columns **must** be tab-separated or comma-separated). Entries must not contain commas or tabs unless they are surrounded by double quotation marks. (Microsoft Excel generally surrounds values containing commas with double quotation marks when saving to a CSV file.)

Finally, the data rows may be preceded by an ID character string starting in column 1 that distinguishes this line as data. To make use of this feature check the "Data rows have ID string" box on the Open dialog, and enter the text (up to 8 characters) in the corresponding box. This string is preserved from one session to the next, and once you enter the appropriate string you will be able

to drag-and-drop these files onto DPlot without the need to check this box each time.

### Example (data row ID string):



If the title line(s) or column headings contain commas, the text to be used should start and end with a double quotation mark (").

Files saved by Microsoft Excel as "comma-separated values" can be read into DPlot using this option.

#### Campbell Scientific datalogger files

For the most part, Campbell Scientific datalogger files are handled identically to CSV and other multiple-column text files, with a couple of exceptions. DPlot determines that a multiple-column text file is a Campbell datalogger file if:

1) The first seven characters in the file are "TOA5", (including the quotes), and

2) The data is determined to start in line 5, and

3) Lines 2 through 4 contain two times as many quotation marks as there are data columns.

If all of the above tests are met, the file is considered a Campbell Scientific datalogger file. The only significant differences in file handling are:

1) The first title line is taken from the second quoted string in the first line (i.e. "T0A5" is ignored). The second title line is taken from the third quoted string in the first line.

2) Legend entries are taken from the second and third lines in the file. If the entries in the third line are not blank, a comma separator is added between the entry from the second line and the entry from the third line.

3) If the column heading for the second column in the second file line is "RECORD", this column is skipped. Normally the 0-based record number is superfluous. If you **want** the record number to be plotted, you should check the "Pick columns to plot" box.

#### Pick Columns to Plot

Allows you to specify which columns within a multiple-column file to plot. Up to 20 rows from the selected file are displayed in the read-only box on the left side of the dialog box. Each column is preceded by a heading with the column number as interpreted by DPlot. If the column numbers do not match up with what you expected, or if the displayed text does not start with the first data line, then DPlot had difficulty in determining either the number of columns or the start of the data, or both. For more information see the description of file type D in the <u>Open Command</u> Help topic.

By default, DPlot uses the first column of data in multi-column files for the X axis values. If there is

only one column of data, DPlot starts X at 0 and increments by 1 for each additional row. Subsequent columns (or the first column in single-column files) are used for the Y values, each sharing the same X.

To change the column number used for the X axis values, enter a number in the *Use column* _____ for *X Axis* box. This entry cannot be a text column. If this number is 0, then DPlot starts X at 0 and increments X by 1 for each additional row of data values.

Select columns to use for the Y values by checking the appropriate check boxes under *Use these columns for* Y. Note that you cannot select a column to serve as both the X Axis values and the Y values for a curve, nor can you select text columns (anything other than numbers, currencies, dates, times, or date-time pairs).

Alternatively, if the file contains alternating X,Y columns, as in:

X(1,1)	Y(1,1)	X(1,2)	Y(1,2)
X(2,1)	Y(2,1)	X(2,2)	Y(2,2)
X(3,1)	Y(3,1)	X(3,2)	Y(3,2)

etc., then you should check the box labelled *Alternating X,Y columns*. In this case DPlot will disable the even-numbered (Y) columns. Checking an X column will automatically cause the corresponding Y column to be checked. *Alternating X,Y columns* cannot be selected if the total number of columns in the file is an odd number, **or** if the file contains any non-numeric columns.

#### Labels

For X,Y data files containing 3 (and only 3) columns or for 3D data files containing 4 columns, you may specify that the last column contains point labels. For example this data:

```
0.0, 0.00000000, "X=$X, Y=$Y"
0.1, 0.309016994,
0.2, 0.587785252,
0.3, 0.809016994,
0.4, 0.951056516,
0.5, 1.00000000, "X=$X, Y=$Y"
0.6, 0.951056516,
0.7, 0.809016994,
0.8, 0.587785252,
0.9, 0.309016994,
1.0, 0.00000000, "X=$X, Y=$Y"
1.1,-0.309016994,
1.2, -0.587785252,
1.3,-0.809016994,
1.4,-0.951056516,
1.5,-1.00000000, "X=$X, Y=$Y"
1.6, -0.951056516,
1.7,-0.809016994,
1.8,-0.587785252,
1.9,-0.309016994,
2.0, 0.00000000, "X=$X, Y=$Y"
```

will produce this plot:



E) Unformatted binary (32-bit). Unformatted data files are generally much smaller than their equivalent ASCII files, and read/write operations are generally an order of magnitude faster. DPlot writes the file as a series of 1024-byte records, with the 1st record containing only the number of data points (4 byte integers) for each curve, and subsequent records consisting of 128 x,y pairs each (with possibly fewer than 128 pairs in the last record for each curve). The data for the second and subsequent curves (if present) must start on a 1024-byte boundary.

Note that this method will always produce a file size of at least 2048 bytes for a single curve (or 1024*(1+number of curves)), regardless of how small the original data set is.

X and Y values are saved as 4-byte (32-bit) floating point values. If precision is important (for example your data has hundreds of thousands of points with a relatively small increment between points), consider using type N. See below.

Files **saved** by DPlot using this format will always fill the first 1024-byte record with 0's following the number of points for each curve, which should assist programmers in quickly determining how many curves are present in the file.

If you use the Lahey FORTRAN compiler and you want to read an unformatted file produced by DPlot, you should compile your program with the /D option, which tells the program not to expect the F77L header. Likewise, if your F77L program produces an unformatted file that you want to read with DPlot, the program should be compiled with the /D option.

Other FORTRAN compilers may expect different file formats for unformatted data than that described above.

- F) Binary file produced by Pacific Data Model 9820 recorders, DNA/Bendix format. Records produced in DNA field tests generally follow this format. See your instrumentation personnel. For more information see the <u>Bendix Format Files</u> Help topic.
- G) Binary file produced by Pacific Data recorders, OLD format (pre-1993). For more information see the <u>Pacific Data Recorder Old Format</u> Help topic.
- H) Binary file produced by Pacific Data recorders, NEW format (post-1993). For more information see the Pacific Data Recorder New Format Help topic.

- Nicolet Time Domain files (binary) created with the Nicolet System 400 Digital Oscilloscope. For more information see the Nicolet Waveform File Specification Format Help topic.
- J) Hardened Data Acquisition System (HDAS) files. The HDAS is a self-contained transducer/ recording system developed by Dr. Ray Franco. For more information see the <u>HDAS File Format</u> Help topic.
- K) Random 3D points. ASCII text file containing randomly-spaced 3D points, one X,Y,Z triplet per line. Values may be separated by commas, spaces, or tabs. The file type will result in a new plot window being opened (if the currently active window has a plot). For surface plots, a convex triangular mesh is generated, and each triangle in the mesh is considered planar when drawing contour levels. Areas outside the triangular mesh are not interpolated and not drawn. To delete extraneous triangles see the How To topic <u>"How do I force DPlot to create a triangular mesh of my 3D points that is not convex?"</u>. You can generate a smoother plot after reading one of these file with the <u>Generate Mesh</u> command on the Options menu. Also for surface plots, points with identical X,Y coordinates are removed, preserving the point with the maximum Z value. For 3D scatter plots, all points are preserved. For 3D scatter plots, this file format can contain multiple data sets. Data sets are separated by a single blank line. 3D scatter plots saved by DPlot as CSV files will use this same format.

This file format may optionally contain 3 title lines and X, Y, and Z axis labels preceding the data. If included, these lines should start and end with double-quotation marks. DPlot reads labels in the order 1st title line, 2nd title line, 3rd title line, X axis label, Y axis label, Z axis label. So if you want, for example, to include all axis labels but have only one title line, the 2nd and 3rd lines in the file should be pairs of double-quotation marks. Alternatively, if the X, Y, and Z axis labels are on the same line (such that there are 3 pairs of double quotation marks on this line), DPlot will interpret this line correctly regardless of whether any title lines precede it.

For data files containing 4 (and only 4) columns, you may specify that the last column contains point labels by checking the **Pick Columns to Plot** box. These labels will **only** be drawn in 2D views, not 3D views. Labels should be delineated with "double quotation marks".

- L) File import plugins. The associated listbox shows plugin modules that you elected to install when running the DPlot setup program. File import plugin modules distributed with DPlot are described in the <u>File Import Plugin Modules</u> topic. To develop your own plugin modules for DPlot, see the <u>Plugins</u> for File Import, Export and Data Manipulation topic.
- M) 1D Statistics. ASCII file containing one or more groups of amplitudes. These files produce a boxand-whisker plot or dot graph. The format for this file type is:

```
NumGroups
do i = 1,NumGroups
Label
NumPoints
do j = 1,NumPoints
Amplitude(j,I)
end do
end do
```

N) Unformatted binary (64-bit). Unformatted data files are generally much smaller than their equivalent ASCII files, and read/write operations are generally an order of magnitude faster. DPlot writes the file as a series of 1024-byte records, with the 1st record containing only the number of data points (4 byte integers) for each curve, and subsequent records consisting of 64 x,y pairs each (with possibly fewer than 64 pairs in the last record for each curve). The data for the second and subsequent curves (if present) must start on a 1024-byte boundary.

Note that this method will always produce a file size of at least 2048 bytes for a single curve (or 1024*(1+number of curves)), regardless of how small the original data set is.

X and Y values are saved as 8-byte (64-bit) floating point values. For a (usually) smaller file size but decreased precision, see <u>type E</u> above.

Files **saved** by DPlot using this format will always fill the first 1024-byte record with 0's following the number of points for each curve, which should assist programmers in quickly determining how many curves are present in the file.

If you use the Lahey FORTRAN compiler and you want to read an unformatted file produced by DPlot, you should compile your program with the /D option, which tells the program not to expect the F77L header. Likewise, if your F77L program produces an unformatted file that you want to read with DPlot, the program should be compiled with the /D option.

Other FORTRAN compilers may expect different file formats for unformatted data than that described above.

#### Array sizes

If the current maximum number of points/curve is less than the number required to read the file, DPlot will attempt to re-allocate the X,Y arrays using the power of 2 greater than or equal to the number of points found in the file, and the number of curves specified in the file. If the re-allocation fails, DPlot presents an error message and/or reads only the allowable number of points.

If, after successfully reading a file, the allocated number of points/curve is greater than twice the number required, DPlot re-allocates the X,Y arrays using the power of 2 greater than or equal to the maximum number of points for all curves.

#### Multiple file selection

For other than DPlot files, you may read multiple files into the same window. To select multiple files, hold down the CONTROL key while selecting each of the files to read, or press the SHIFT key to select a range of files.

#### **File termination**

For each of the supported ASCII file formats, the last line in the file should include a carriage return/line feed sequence.

#### Time scale

Input time values for binary files (file types F, G, H, I, and J) are automatically shifted to milliseconds, rather than seconds.

#### 烤 Related macro commands

ColumnsAre FileOpen ForFilesIn UseNameAsLegend

# 4.1.2.1 DPlot File

A standard DPlot file is an ASCII text file containing all of the information DPlot needs to fully reproduce a finished plot. DPlot will also produce compressed DPlot files which contain the same information below but are compressed (and not human-readable) by the <u>ZLIB</u> algorithm.

The first line in the file contains a header that informs DPlot of the file version. The second line contains a keyword that describes what follows. The general form of a DPlot file follows:

DPlot v1.6 keyword nCurves do i = 1 to nCurves keyword-specific data LineStyle(i), SymbolStyle(i) Legend(i) CurveLabel(i)

```
end do
Title(1)
Title(2)
XAxisLabel
YAxisLabel
ScalingCode
LegendX, LegendY|,LegendAlignment|
Other miscellaneous identifiers
For "keyword" = "1D", the file format is:
DPlot v1.6
1D
NumGroups
do I = 1 to NumGroups
   NumPoints
   do j = 1 to NumPoints
      z(I,i)
   end do
   Label
end do
For "keyword" = "3D", the file format is:
DPlot v1.6
3D
x_lowerlimit, y_lowerlimit, x_upperlimit, y_upperlimit
NumX. NumY
do I = 1 to NumX+1
   do j = 1 to NumY+1
      Z(I,j)
   end do
end do
```

followed by Title(1), Title(2), etc. NumX and NumY are the number of grid intervals in the X and Y directions. The total number of Z values is then (NumX+1)*(NumY+1). Z values may be on the same or separate lines.

```
For "keyword" = "3DR" the file format is:

DPlot v1.6

3DR

NumPoints

do I = 1 to NumPoints

x(i), y(i), z(i)

end do
```

followed by Title(1), Title(2), etc.

If the maximum number of curves (set with **Resize arrays**) is less than the number of curves specified in the file, DPlot will automatically adjust the size of the X and Y arrays so that: 1) the maximum number of curves is equal to the number of curves specified in the file; and 2) the maximum number of points is greater than or equal to the maximum number of points for any one curve. This feature may require DPlot to read the file several times before finding the minimum array sizes.

The file header is a recent addition to the DPlot file format that assists DPlot in immediately identifying a file. Older DPlot files will continue to work with DPlot.

See also: Resize arrays

# 4.1.2.2 Bendix Format Files

This file format is frequently used in field tests with the Pacific Data Model 9820 recorder.

Bendix format files consist of a header block followed by binary data. The header structure used in these files is defined below:

typedef st	<pre>ruct tagHEADER {</pre>
short	Model;
short	Chan;
short	Module;
short	Config;
short	NumPost;
short	<pre>Profile[15];</pre>
char	<pre>Eventname[40];</pre>
double	Trigger;
short	CounterLoop;
long	StartTime;
char	Status[8];
short	Memory;
char	<pre>Engineering[14];</pre>
float	Baseline;
float	Calibration;
char	AutoX[8];
float	LowerLimit;
float	UpperLimit;
char	AutoY[8];
float	PlotStart;
float	FinishPlot;
short	PostSegs;
short	SEGM[16];
char	TriggerStatus[8];
char	<pre>ProtectSwitch[8];</pre>
char	<pre>ProtectFlag[8];</pre>
char	ErrStatus[8];
char	FilterValue[8];
char	Measurement[14];
char	StampTime[8];
char	StampDate[10];
char	Bunker[14];
cnar	AmpGain[8];
long	DECTrigger;
long	DECDUMMY;
float	VoltsLSB1;
char	VUILSLSBZ,
char	CableNumber[14],
char	TPoyNumber[14]
char	CaceNumber[14]
CHAL	Gayenumer [14]/

```
GageType[14];
char
char
        CableType[14];
char
        TermResistor[14];
char
        OperName[14];
        NumberChannels;
short
float
        VoltsLSB3;
float
        VoltsLSB4;
short
        SampleClockCode;
short
        CoarseOffset;
        FineOffset;
short
short
        Dummy[296];
        DataBaseVersion[2];
char
```

} HEADER;

The Bendix format is used for a variety of Pacific Data recorder models. However, DPlot only supports files produced with the model 9820 recorder (Model=9820 in the Header block).

Calibration data (if present) consists of 1024 2-byte values located between offsets 1024-3071 in the file. The raw data for Model 9820 files starts at offset 3072 in the file. Raw data is organized into 15 segments of either 4096 or 8192 2-byte integers each. Each segment can have a unique time step. The time step (in microseconds) for a given segment is determined from:

dT = 1.0 << (16-(Profile[segment] & 15);</pre>

The raw data and calibration data are translated to engineering units by:

Y = float(Volts-2047)*VoltsLSB1;

where Volts represents the raw data, and VoltsLSB1 is from the file header.

If calibration data is present (indicated by a non-zero Calibration in the header), the output amplitudes are adjusted by:

Y = (Y - CalBase)*Calibration /(CalCal-CalBase);

where CalBase is the average of the 1st and 3rd blocks of 256 calibration values, and CalCal is the average of the 2nd and 4th blocks of 256 calibration values.

# 4.1.2.3 Pacific Data Recorder Old Format

This file format is frequently used in field tests with Pacific Data recorders.

These files consist of a header block followed by data representing measured voltage. The header structure used in these files is defined below:

```
typedef struct tagHEADER {
```

```
char
      taq name[8];
char
       units[4];
char
      channel_description[20];
       channel location[20];
char
char
       not used 12[12];
char
       file test name[40];
char
       file_creation_date_and_time[24];
short
       trigger timeout value;
short
       number of pretrigger segments;
short
       number of posttrigger segments;
short
       sample_rate_profile[16];
short
       configuration_word;
       number_of_raw_data_files_in;
short
```

```
auto_calibration_procedure;
short
        auto_calibration_order;
short
        internal_trigger;
short
char
       not_used_80[80];
float
       convert[4];
float
       trigger_level;
float
        correlation_data[48];
float
        calibration_conversion[10];
float
        reserved;
```

```
} HEADER;
```

Files produced using this format are always either 263,936 bytes or 526,080 bytes (Modar model 5700 only) in size. The header information starts at offset 1024 from the start of the file. Information preceding the header is not used by DPlot. Raw data is organized in 16 segments of either 16384 bytes or 32768 bytes (Modar model 5700 only) beginning at offset 1792 in the file. Each segment may have a different time step, which is determined (in microseconds) by:

```
float time_mlpy[16] { 32768., 16384., 8192., 4096.,
                       2048., 1024., 512., 256.,
                        128.,
                                 64., 32.,
                                               16.,
                                  4.,
                                         2.,
                          8.,
                                               1. };
float Modar_time[16] {100000., 50000., 20000., 10000.
                        5000., 2000., 1000.,
                                                 500.,
                         200.,
                                 100.,
                                          50.,
                                                  20.,
                                                 1. \};
                          10.,
                                   5.
                                           2.,
for(i=0;i<=15;i++) {</pre>
   if( i < Header.number of pretrigger segments )
     k = 0;
   else
      k = i - Header.number of pretrigger segments + 1;
   if( ModarFile )
      TimeStep[i] = time_mlpy[Header.sample_rate_profile[k]];
   else
      TimeStep[i] = Modar time[Header.sample rate profile[k]];
}
```

Raw data is translated to engineering units by the cubic polynomial:

```
volts = float( RawData ) / 3276.8;
Y = ( ( Header.Convert[3]*volts +
    Header.Convert[2] )*volts +
    Header.Convert[1] )*volts +
    Header.Convert[0];
```

# 4.1.2.4 Pacific Data Recorder New Format

This file format is frequently used in field tests with Pacific Data recorders.

These files consist of a header block followed by data representing measured voltage. The header structure used in these files is defined below:

```
typedef struct tagHEADER {
  char
          taq[8];
                              /* Channel name
                                                             * /
                             /* Channel description
                                                             */
  char
          desc[20];
                             /* Gage location
  char
          loct[20];
                                                             */
                              /* Engineering units
                                                             */
          units[14];
  char
                              /* Time and date channel
  char
          datetime[19];
```

```
*/
                               was saved
char
        sernum[13];
                            /* Gage serial number
                                                           */
        model[15];
                                                           */
char
                            /* Gage model
        range[8];
char
                            /* Gage range
                                                           */
        cable[5];
                            /* Instrumentation cable
char
                                                           * /
                               number
                            /* Test name
char
        testname[40];
                                                           */
char
        testloc[20];
                            /* Test location
                                                           */
char
        operator[20];
                           /* Instrumentation technician */
                           /* TDR number
short
        channel;
                                                           */
        caltype;
short
                            /* Automatic (1) or
                                                           * /
                               Simulated (2) Cal
                            /* Internal trigger enabled=-1,
short
        trigenable;
                               disable=0
                                                           * /
                            /* # of pretrigger segments
short
        numpre;
                                                           */
                           /* # of posttrigger segments
short
        numpost;
                                                           */
short
        rate[15];
                           /* Rate in microseconds for
                                                           * /
                               each segment
                            /* TDR memory size in KB
short
        memsize;
                                                           */
float
        val1;
                            /* Cal baseline in engineering
                               units
float
        val2;
                            /* Cal step in engineering
                               units
                                                           * /
float
        svolts1;
                            /* Cal baseline in volts
                                                           * /
float
                           /* Cal step in volts
                                                           */
        svolts2;
float
        poly1;
                            /* Conversion polynomial
                                                           * /
                               offset
float
        poly2;
                            /* Conversion polynomial scale
                               factor
                                                           * /
float
                            /* Trigger level in
        triglevel;
                                                           * /
                               millivolts
float
                            /* Gage CA
                                                           */
        ca;
float
        re;
                            /* Gage RE
                                                           */
float
                            /* Gage RX
                                                           * /
       rx;
                           /* Gage RG
float
       rg;
                                                           * /
                           /* Cable line resistance
                                                           */
float
        tworl;
                          /* Engineering prediction
float
       pred;
                                                           */
                          /* Actual cal resistor used
float
       rc;
                                                           */
double strttotrig;
                          /* Start to trigger time
                                                           */
short
        prate;
float
        gain;
float
        sensitivity;
char
        projectname[40];
char
        dummy[1690];
```

} HEADER;

Files produced using this format are always 264,192 bytes in size. The header block (shown above) begins at the start of the file, and is followed (at offset 2048) by 16 segments of 8192 2-byte integers representing voltages (the last block of 16384 bytes is not used by DPlot). The time step for each segment (in microseconds) is taken from the rate member of the header.

Raw voltages are translated to engineering units by:

Volts = float(RawData)/32768.; Y = Header.poly2 * Volts + Header.poly1;

## 4.1.2.5 Nicolet Waveform File Specification Format

This file format is created with the the Nicolet System 400 Digital Oscilloscope.

These files consist of a header block followed by data representing measured voltage. The header structure used in these files is defined below:

```
typedef struct tagHEADER {
          Nic_id0[2];
                                    /* 3 indicates Intel
  char
                                       byte ordering
                                                          */
          Nic id1[2];
                                    /* 1 indicates time domain
  char
                                      (rather than frequency */
  char
          Nic id2[2];
  char
          User_id[2];
  char
          Header size[12];
          File_size[12];
  char
  char
          File format version[12];
  char
          Waveform title[81];
  char
          Date_year[3];
  char
          Date_month[3];
  char
          Date_day[3];
  char
          Time[12];
  char
                                    /* Number of raw data values */
          Data_Count[12];
  char
          Vertical zero[12];
          Vertical_norm[24];
  char
  char
          User vertical zero[24];
  char
          User vertical norm[24];
  char
          User vertical label[11];
  char
          User_horizontal_zero[24];
  char
          User_horizontal_norm[24];
  char
          User_horizontal_label[11];
  char
          User_Notes[129];
  char
          Audit[196];
          Nicolet_Digitizer_Type[21];
  char
  char
          Bytes_per_data_point[3]; /* Must be 2 for DPlot */
  char
          Resolution[3];
  char
          Forward link[81];
          Backward_link[81];
  char
  char
          Process flag[3];
  char
          Data_compression[3];
  char
          Number_of_segments[12];
  char
          Length_of_each_segment[12];
  char
          Number_of_timebases[12];
  char
          Reserved 1[156];
  char
          Length_of_zone_1[12];
  char
          Horiz norm zone 1[24];
  char
          Horiz_zero_zone_1[24];
                                   /* Must be null char for DPlot */
  char
          Length_of_zone_2[12];
  char
          Horiz_norm_zone_2[24];
  char
          Horiz zero zone 2[24];
          Length_of_zone_3[12];
                                   /* Must be null char for DPlot */
  char
  char
          Horiz_norm_zone_3[24];
```

```
char Horiz_zero_zone_3[24];
char Reserved_2[332];
char End_of_HDELTAS[1];
char End_of_readable_file[1];
} HEADER;
```

Raw data starts immediately after the header, or at the offset specified by the Header_size member of the header. Time values and Engineering units for the i'th data point are determined by:

```
Time = ((float(i)*Horizontal_Norm) + Horizontal_Zero ) *
    User_Horizontal_Norm + User_Horizontal_Zero;
Y = ((float(RawData)-Vertical_Zero ) * Vertical_Norm ) *
    User_Vertical_Norm + User_Vertical_Zero;
```

where Horizontal_Norm, Horizontal_Zero, User_Horizontal_Norm, User_Horizontal_Zero, Vertical_Norm, Vertical_Zero, User_Vertical_Norm, and User_Vertical_Zero have been translated from the corresponding character strings in the header to floating point numbers.

# 4.1.2.6 HDAS File Format

The HDAS (Hardened Data Acquisition System) is a self-contained transducer/recording system developed by Dr. Ray Franco of the Instrumentation Services Division at the US Army Engineer Research and Development Center in Vicksburg, Mississippi.

These files consist of voltage samples followed by a footer describing the test data. The footer structure used in these files is defined below:

```
typedef struct tagFOOTER {
  char
       Gain[20];
  char
          Sensitivity[30];
          Excitation[20];
  char
  char
          SamplingPeriod[30];
          CalResistor[20];
  char
  char
         YAxisUnits[6];
  char
          SiteLocation[30];
  char
       GaugeSerialNumber[40];
  char retriggerLocation[6];
  char
          FullScaleAD[20];
  char
          XAxisScaler[20];
         YAxisScaler[20];
  char
         YAxisZeroOffset[20];
  char
         XAxisZeroOffset[10];
  char
  char
          XAxisUnits[6];
  char
         CA[20];
  char
          Rq[20];
          Reserve[138];
  char
```

```
} FOOTER;
```

Files produced using this format are always 262,620 bytes in size. Calibration data begins at the start of the file. Calibration data is read from the first and fourth blocks of 1024 2-byte values in the file. Raw data, consisting of voltage measurements, begins at offset 8192 in the file. The time step, in microseconds, is read from the SamplingPeriod member of the footer. Time and engineering unit values for the i'th point are determined by:

```
Time = (-XAxisZeroOffset+4095+i)*SamplingPeriod;
Y = ( float(RawData & 0x07FF) - YAxisZeroOffset ) *
```

```
(CA/(CalResistor+Rg) / (CalTop-CalBottom)
```

where SamplingPeriod, XAxisZeroOffset, YAxisZeroOffset, CA, CalResistor, and Rg are the floating point equivalents of the corresponding character strings in the footer, and CalTop and CalBottom are the average calibration values read from the 1st and 4th blocks of 1024 calibration values, respectively.

#### 4.1.2.7 File Import Plugin Modules

Plugin modules for file import are separate, optional components for DPlot. Programmers interested in developing their own plugin modules should read the <u>Plugins for File Import, Export and Data</u> Manipulation topic. File import plugins distributed with the full version of DPlot are described below.

## **AMO Transient Recorder files**

Supports AMO Saturn Transient Recorder files.

#### **ArcView Shapefile**

Reads ArcView Shapefiles. A shapefile is a popular geospatial vector data format for geographic information systems software developed by <u>ESRI</u>. This plugin expects to find both a .SHP main file and .SHX index file with the same filename in the same directory. The plugin will automatically set *Degrees, Minutes* number formats for both the X and Y axes, insert nonsense Y=100 points between land masses, and set the scaling to Mercator projection, which in turn forces the *Amplitude Limits* setting on with limits of +/- 85 degrees (so that those previous nonsense points are ignored, and separate land masses will be drawn separately). If the file contains more than one entity and fewer than 100, you are given the option of reading the entire file into one curve or using one curve per entity.

Although DPlot will display longitudes as expected (-200° is displayed as 160°, for example), on input the data will be drawn with decreasing x values (longitudes) to increasing x values from left to right. Therefore if your map crosses longitude 180° and your input longitudes are positive in the Eastern hemisphere, negative in the Western hemisphere (as is standard practice), you will need to modify the longitudes for the map to be drawn correctly; otherwise you will get the entire range -180 to 180 degrees on the X axis. This change is easily accomplished with the <u>Operate on X</u> command on the Edit menu with X=IF(X>0,X-360,X). For example in this map:



the westernmost Aleutians are stored in the shapefile as positive longitudes. Without the edit mentioned above, the map would be drawn with the majority of the state at the left extreme of the plot and the westernmost Aleutians at the right extreme.)

#### Bitmap to 3D

This plugin maps pixel values of 8-bit images to Z values in a surface plot. Non 8-bit images will be converted to 8-bit gray scale. Generally, gray scale images are stored with black as color 0, white as color 255. The plugin supports the most common bitmap formats: Windows bitmaps (BMP), Graphics Interchange Format (GIF), JPEG, Portble Network Graphics (PNG), Tagged Image File

Format (TIF), and Targa (TGA) files.

NOTE: Bitmaps with distinct color bands (as you'd get with a screen capture of a DPlot plot) will result in stairstepped contour plots. For best results use bitmap images with a smooth gradation of shades.

Example:



# **Gould DAT files**

Supports 8- and 16-bit data produced by most data loggers from Gould Instrument Systems (now defunct).

## Hi-Techniques Win600 DAT files

Reads files produced by Hi-Techniques data acquisition systems.

### LeCroy WavePro files

Reads files for the Lecroy WavePro line of oscilloscopes.

#### Raw data

Reads a file containing one or more data sets stored in the native binary format for the data type. Date consists of amplitudes (Y values) only. The plugin will generate X values starting at 0 with spacing determined by the "sample rate" entry. Data may be 1-, 2-, or 4-byte signed or unsigned integers or 4-byte (single precision) or 8-byte (double precision) floating point values in IEEE format. Both Intel little endian and Motorola big endian byte orders are supported. (Generally, software written for Apple computers and UNIX/Linux operating systems produces output with Motorola byte-ordering, while Windows programs generally use Intel byte order.)

#### **Tektronix WFM files**

Reads WFM files produced by most of the oscilloscopes from Tektronix.

#### WAV Audio

Reads Windows WAV files, 8- or 16-bit, mono or stereo, any valid recording rate. Input amplitudes are normalized to +/- 1.0. Stereo files result in 2 curves, mono in one curve.

# Z Table

This plugin expects to read tabulated z values from an ASCII text file in the form:

 $z(x_1,y_1), z(x_2,y_1),..., z(x_n,y_1)$  $z(x_1,y_2), z(x_2,y_2),..., z(x_n,y_2)$ 

 $z(x_1, y_m), z(x_2, y_m), ..., z(x_n, y_m)$ 

where n is the number of columns (x values) and m is the number of rows (y values). Rows and columns are assumed to be evenly spaced between the specified. Data values may be separated by commas, spaces, or tabs.

# 4.1.3 Append command

Adds data from a file to the currently active document window. Valid file formats are described under the <u>Open command</u> Help topic.

# Related macro commands

**FileAppend** 

# 4.1.4 Save command

Use this command to save the active document in the DPlot file format. When you save a document for the first time, DPlot displays the Save As dialog box so you can name your document. If you want to change the name, location, or format of an existing document before you save it, choose the <u>Save As</u> command.

If this file was previously saved as a compressed DPlot file, it will again be saved in compressed form without your interaction. Likewise, if the file was uncompressed, it will be saved uncompressed. If you want to save an uncompressed document as compressed or vice versa, use the <u>Save As</u> command. DPlot uses the <u>ZLIB</u> algorithm to compress files. Though the compression algorithm is identical to the ZIP format, compressed DPlot files **cannot** be opened by WinZip[®] or similar archive utilities. They can, of course, be opened by DPlot.

# Shortcut: Click on the toolbar. CTRL+S Related macro commands FileSaveAs SetFilename

# 4.1.5 Save As command

Saves either the entire plot, or a single curve's data, depending on the format chosen.

#### Note:

- 1) To save a DPlot file with its existing name, format, and location, use the Save command.
- 2) If one of the file formats described below does not appear in the "Save as type" box, it is simply because that file format is inapplicable for the current graph. For example you cannot save 3D views of 3D or 4D data as a metafile (simply because the file would be enormous).

#### **DPlot file**

Saves all data in addition to the title, axis labels, legend, and other formatting information.

#### **Compressed DPlot file**

Identical to the above, but compressed using the <u>ZLIB</u> algorithm. As such the file is not humanreadable and may not be edited or read in a text editor (or by WinZip[®] or similar archive utilities). In general files compressed using this method will have a compression ratio from 2.5:1 to 4:1. Larger files may achieve 12:1 or higher, and extreme cases with long runs of identical values might achieve 400:1, though this is rare. The compression is lossless; all information will remain intact. Compressed DPlot files may be opened with DPlot or DPlot Viewer, versions 2.2 or later.

#### ASCII file (1 curve only)

```
Saves one curve to a file with the following form: nPoints x(1) \quad y(1) \\ x(2) \quad y(2)
```

x(nPoints) y(nPoints)

If you use a filename extension of .csv, the columns will be comma-separated. Otherwise the columns will be tab-separated.

#### **Comma-separated values**

For 2D plots, saves all curves to a file with the following form:

where N is the number of curves and M is the number of points per curve.

For XY plots this option is only available if either 1) all curves share the same X values and all curves have the same number of data points or 2) all curves are sorted in increasing X order. In the latter case blank entries (two successive commas) will be written for any curve that does not have a data point at the X for the line being written.

The title lines are written to the file only if present.

If the plot consists of one curve only, then the column label line is written as:

```
"X Axis Label","Y Axis Label"
or
"X Axis Label","Legend 1"
```

if no Y axis label is present.

For the first column, if no X axis label is present then DPlot will use "Time" for *Time of day* <u>number</u> format, "Date" for *Calendar Date*, "Date/Time" for *Date and Time*, or "" for other formats.

This file format may be opened directly in Microsoft Excel and opened in DPlot as file type D, "Multiple columns". You may also drag these files from Explorer onto the DPlot window to open them.

For 3D plots, saves all points to a file in this form:

For 3D/4D scatter plots with multiple data sets, each set will be written sequentially and separated by

## a blank line.

This file format may be opened directly in Microsoft Excel and opened in DPlot as file type K, "Random 3D points".

#### **Tab-separated values**

Identical in all respects to comma-separated values, except that the data columns are separated by tab characters rather than commas. As with CSV files, this file format may be opened directly in Microsoft Excel (after answering prompts concerning the delimiter) and opened in DPlot as file type D, "Multiple columns" for XY plots or file type K, "Random 3D points" for 3D data.

# Unformatted binary (32-bit)

Saves the data for all curves as a series of 1024-byte records. Unformatted data files are generally much smaller than their equivalent ASCII files, and read/write operations are generally an order of magnitude faster. DPlot writes the file as a series of 1024-byte records, with the 1st record containing only the number of data points (4 byte integers) for each curve, and subsequent records consisting of 128 x,y pairs each (with possibly fewer than 128 pairs in the last record for each curve). The data for all curves starts on a 1024-byte boundary.

Note that this method will always produce a file size of at least 1024*(1+number of curves), regardless of how small the original data set is.

Files saved by DPlot using this format will always fill the first 1024-byte record with 0's following the number of points for each curve, which should assist programmers in quickly determining how many curves are present in the file.

#### Unformatted binary (64-bit)

Identical to the above, but data is saved as 8-byte (64-bit) values.

Saves the data for all curves as a series of 1024-byte records. Unformatted data files are generally much smaller than their equivalent ASCII files, and read/write operations are generally an order of magnitude faster. DPlot writes the file as a series of 1024-byte records, with the 1st record containing only the number of data points (4 byte integers) for each curve, and subsequent records consisting of 64 x,y pairs each (with possibly fewer than 64 pairs in the last record for each curve). The data for all curves starts on a 1024-byte boundary.

Note that this method will always produce a file size of at least 1024*(1+number of curves), regardless of how small the original data set is.

Files saved by DPlot using this format will always fill the first 1024-byte record with 0's following the number of points for each curve, which should assist programmers in quickly determining how many curves are present in the file.

#### Windows Metafile

Saves the entire plot in the Windows placeable metafile format. Placeable metafiles may be imported into a number of word processors and spreadsheet applications, or may be converted to another file format by many popular graphics conversion programs. If the "Set Plot Size" checkbox on the <u>Extents/Intervals /Size</u> dialog box is in effect, the plot will be sized such that the box bounding the plot has the dimensions specified by that command. Otherwise the dimensions for the entire metafile will match those specified with the <u>Copy Dimensions</u> command.

DPlot will not read a metafile.

DPlot will **not** save a metafile of a 3D view or a 2D view of a surface plot with the "Type" set to "Shaded bands" or "Both lines and shades".

#### **Enhanced Metafile**

Saves the entire plot in the enhanced metafile format. Enhanced metafiles may be imported into a number of word processors and spreadsheet applications, or may be converted to another file format by many popular graphics conversion programs. Enhanced metafiles contain scaling information not found in Windows metafiles, and may not exhibit some of the problems in other applications (e.g. MS PowerPoint) that are present with placeable metafiles. The enhanced format is generally superior to the placeable metafile format.
DPlot sizes the enhanced metafile as described above under Windows Metafile.

DPlot will not read an enhanced metafile.

DPlot will **not** save an enhanced metafile of a 3D view or a 2D view of a surface plot with the "Type" set to "Shaded bands" or "Both lines and shades".

## **Bitmap images**

These file formats may or may not appear on this dialog, depending on whether you have installed the corresponding plugins. After selecting one of these formats you will be presented with the <u>Image</u> <u>Properties</u> window, in which you can control the size, bit depth, and several other settings.

## **BMP** Picture

Saves plots to 8-bit uncompressed, 8-bit run-length-encoded (RLE), or 24-bit uncompressed bitmaps. The 8-bit RLE format is generally suitable for most DPlot graphs and produces considerably smaller files than the other BMP options. However not all applications will support RLE bitmaps. 24-bit uncompressed files will be **much** larger than any of the other image formats but might be useful when, for example, using DPlot-produced bitmaps as frames in a movie.

### **Graphics Interchange Format**

The image is limited to 256 colors. <u>LZW compression</u> is used. For XY plots this is generally not a limitation, but may result in loss of detail in 3D images with many colors.

## JPEG Picture

Saves a JPEG image. For quality settings other than "Lossless", the quality generally isn't good relative to lossless formats (BMP, PNG, TIFF).

### Portable Network Graphics (PNG)

The best option for saving bitmap images if your target application supports the format. PNG images are lossless and images of DPlot graphs will generally compress to a smaller size than any other format.

## Tagged Image File Format (TIF)

The TIFF plugin uses <u>LZW compression</u> on all images. This generally results in compression better than JPEG (particularly for 8-bit images) without losing information.

For all image exports, if the width or height values are set to 0 **and** the resolution is set to 100 or less dpi, the saved image will essentially be a screenshot. If greater than 100 then font sizes, line widths, the plot size and other dimensions as seen on the display will be scaled up by *resolution*/ (100 dpi), so that the saved image will have the same proportions as the graph shown on your display. For best results with a resolution setting greater than 100, always specify the plot size with the <u>Extents/Intervals/Size</u> menu command (or via macro with the <u>Size</u> command). Otherwise you risk generating an *Insufficient memory* error for high resolution settings.

If you are saving to an 8-bit color depth **and** your image has a colored background, you will generally get superior results if you check the *Improved color quantization* box. For large images (large in this case defined as wider than 3500 pixels), the *improved* quantization method will be used regardless of your selection here, primarily because the default method performs poorly with large images.

**NOTE:** For images with many unique colors and/or <u>antialiased lines and symbols</u> and for 3D views of 3D/4D data in which one of the light source models is used, you will get better results saving the image as 24- rather than 8-bits. If file size is paramount you can then use a dedicated image editor like <u>Adobe Photoshop</u> or <u>Paint Shop Pro</u> to reduce the color depth, with better results than you will get by saving to an 8-bit image in DPlot.

## WAV Audio

Support for WAV export is included in the licensed version of DPlot only, and will only appear on the *Save As* dialog if you elected to install this module at installation-time.

Saves one curve to a monotone wave file or two curves to a stereo wave file. For stereo files, the first curve is saved to the left channel; the second curve is saved to the right channel. Amplitudes are assumed to range from -1.0 to 1.0; values outside that range are clipped. The Windows Sound Recorder application can play files saved using this option.

## 烤 Related macro commands

<u>FileSaveAs</u> <u>SetFilename</u> <u>SetImageCrop</u> SetPluginImageDims

## 4.1.5.1 Image Properties

This window is presented when you elect to save a graph as an image file (BMP, GIF, JPG, PNG, or TIF) using the <u>Save As</u> menu command.

#### Width and Height

Enter the desired width and height of the image, in pixels. If the width or height are set to 0 **and** the *Resolution* setting is 100 or less dpi, the saved image will essentially be a screenshot. If both width and height are non-zero then the output image will be these dimensions **unless** the *Resolution* setting is greater than 100 dpi or if the *Crop white space* checkbox is checked. If *Resolution* is set to greater than 100 dpi, then the image dimensions (along with font sizes, line widths, symbol sizes, etc.) will be scaled by *resolution*/(*screen resolution*). For most Windows systems the display resolution is 96 dpi. If *Crop white space* is checked, the image will be the above dimensions minus any extraneous white space, with a 10-pixel white space border.

## Crop white space

If checked, all but a 10-pixel white space border will be cropped from the image. If unchecked, the specified dimensions (subject to the *Resolution* setting) will be used and no cropping is performed.

#### Resolution

If the resolution is set greater than 100 then font sizes, line widths, the plot size and other dimensions as seen on the display will be scaled up by *resolution/(screen resolution)*, so that the saved image will have the same proportions as the graph shown on your display.

For best results with a resolution setting greater than 100, always specify the plot size with a <u>Size</u> command or the <u>Extents/Intervals/Size</u> menu command. Otherwise you risk generating an *Insufficient memory* error for high resolution settings.

#### Color depth

Choose between 8-bit (256 colors), 24-bit (roughly 16 million colors), and 32-bit PNG with an alpha channel. GIF images are limited to 8-bit, JPG images are limited to 24-bit (and so these options are disabled for those formats).

For 32-bit PNG format, the plot background color will be set to transparent within the image. The <u>antialiasing</u> setting will be temporarily turned off if it was on, since removal of a single unique color would otherwise result in ugly artifacts. DPlot will temporarily turn off your system's "Smooth screen fonts" and/or ClearType settings, for the same reason that antialiasing is turned off. This format contains the same information as 8-bit GIF images with "Transparent GIF" turned on, but you may get better results when resizing these images than you would with GIFs.

**NOTE:** For images with many unique colors and/or <u>antialiased lines and symbols</u> and for 3D views of 3D/4D data in which one of the light source models is used, you will get better results saving the image as 24-bit PNG rather than 8-bits. If file size is paramount you can then use a dedicated image editor like <u>Adobe Photoshop</u> or <u>Paint Shop Pro</u> to reduce the color depth, with better results than you will get by saving to an 8-bit image in DPlot.

## RLE-encoded (BMP only)

Only applicable to 8-bit BMP images. The 8-bit RLE format is generally suitable for most DPlot graphs and produces considerably smaller files than uncompressed bitmaps. However not all applications will support RLE bitmaps.

#### Improved color quantization

Only applicable for 8-bit images. If you are saving to an 8-bit color depth **and** your image has a colored background, you will generally get superior results if you check this box. For large images (large in this case defined as wider than 3500 pixels), the *improved* quantization method will be used regardless of your selection here, primarily because the default method performs poorly with large images.

### **Transparent GIF**

Only applicable for GIF images. If checked, the plot background color will be set to transparent within the image. The <u>antialiasing</u> setting will be temporarily turned off if it was on, since removal of a single unique color would otherwise result in ugly artifacts. Depending on your purpose you might also set the <u>number format</u> for the X and Y axes to "None" and delete all text entries that are not essential (title and axis labels, etc.). DPlot will temporarily turn off your system's "Smooth screen fonts" and/or ClearType settings, for the same reason that antialiasing is turned off. This means that text entries in the image will suffer a bit from the jaggies.

#### JPEG quality

Only applicable for JPEG images. The "poor" setting produces the smallest files, though the quality will generally not be acceptable. "Lossless" has no visual artifacts typical of JPEG files, but the file size will be much larger than with any other setting. When DPlot is first installed, the default quality setting is "Good" (4th of the 5 settings).

## 烤 Related macro commands

<u>SetPluginImageDims</u>

## 4.1.6 Close command

Use this command to close the window containing the active document. DPlot suggests that you save changes to your document before you close it. If you close a document without saving, you lose all changes made since the last time you saved it. You can also close a document by using the Close command on the document Control menu.

To close all open document windows, use the Close All command (hold down the SHIFT key and select the File menu).

## Shortcut:

Double click the document Control menu icon.

CTRL+F4

## 烤 Related macro commands

**FileClose** 

## 4.1.7 Get Preferences command

Retrieves plot settings from a previously saved file.

### **Dialog Box Options**

#### Look in

Lists the available folders and files. To see how the current folder fits in the hierarchy on your computer, click the down arrow. To see what's inside a folder, click it.

The box below shows the folders and files in the selected location. You can also double-click a folder or file in that box to open it.

To open the folder one level higher, click *b* on the toolbar.

#### File name

Provides a space for you to type the name of the file.

You can use * as a wildcard. For example, you can type *.* to see a list of all files. You can also type the full path of a file. For example, you can type **c:\myplots\data.set**. Or, if you have used a long filename, you might type **c:\myplots\settings for report plots**.

### Files of type

Lists the types of files to display.

This is useful for narrowing the list of files displayed to only those files you're interested in.

## 烤 Related macro commands

**GetPreferences** 

## 4.1.8 Save Preferences command

Saves plot settings (line and symbol styles and sizes, scaling, etc.) to a file.

### Dialog Box Options

## Save in

Lists the available folders and files. To see how the current folder fits in the hierarchy on your computer, click the down arrow. To see what's inside a folder, click it.

The box below shows the folders and files in the selected location. You can also double-click a folder or file in that box to open it.

To open the folder one level higher, click is on the toolbar.

### File name

Provides a space for you to type the name of the file.

You can use * as a wildcard. For example, you can type *.* to see a list of all files. You can also type the full path of a file. For example, you can type **c:\myplots\data.set**. Or, if you have used a long filename, you might type **c:\myplots\settings for report plots**.

#### Save as type

Specifies the filename extension of the file you are saving.

**NOTE:** If the <u>Automatically read preference file based on filename</u> option under the General command on the Options menu is checked, then when opening a data file for a new document DPlot will check for the existence of a preferences file (previously saved with this command) with the same filename and extension of ".set", first in the folder where the data file resides, then in the default preferences folder (%APPDATA%\DPlot\Preferences). If found, that preferences file will be processed. This might be a handy option if you have data files which frequently change, but you don't want to go through the same formatting steps each time you open that file. This option is not applicable to DPlot files, which by definition contain all of the formatting options that would normally be saved to a preferences file. When opening multiple files, this option is only applicable to the first file selected.

烤 Related macro commands

**SavePreferences** 

## 4.1.9 Print command

The **Print** command allows you to alter the current printer setup and/or print the currently active plot or a selection of open plots.

## Shortcut:



CTRL+P

The initial printer settings are identical to those set with the **Control Panel:Printers** application. Changes in printer settings are preserved only during the current DPlot session; DPlot does not make permanent changes to the **Control Panel** settings, nor does it save printer settings for subsequent DPlot sessions.

## **Dialog box Options**

### **Multiple**

Select the "Multiple" button if you want to print multiple documents on the same page. Placement of documents on the page is controlled by the order in which the documents are selected for printing. The spacing between documents is controlled by the value entered in the "Space between" box.

### **Print Quality/Speed Considerations**

### TrueType Fonts

If you use TrueType fonts in your plot, it may be necessary to check the "Print TrueType as Graphics" check box in the **Print Setup** or **Print Options** dialog box. Some printer drivers do not handle rotated TrueType text correctly unless it is converted to a graphic before printing.

## **Gray Scale and Color**

For black and white printers, DPlot simulates gray scales and colors with a dithered pattern. In order for the curve to be printed as expected, the line widths for the curves must be greater than 0 (a single printer dot). The line width may be changed with the <u>Symbol/Line Styles</u> command under the Options menu. The color associated with each curve may be changed from the Colors menu. Alternatively, you can select the **Color>Print B&W** command on the **Options** menu to print all curves and grid lines in black.

## **3D Contour Plots**

The quality of printed 3D contour plots is very highly dependent on printer settings. Though it is of course not possible to describe optimal printer settings for every available printer in this document, a description of settings for the Epson Stylus family of color printers may assist you in choosing the best settings for your particular printer. The most obvious printer setting that has an effect on print quality is resolution, though you'll likely see little difference in resolutions above 720 dpi. Much more important are the dithering and color balance options. Choose either "fine dithering" or "error diffusion" rather than "coarse dithering", and select "photo-realistic" color balance rather than "automatic".

### Large Data Sets

Printing large data sets can take a long time, depending on what options you have selected and of course on the speed of your processor. In general, solid lines are printed faster than other line styles, and line widths of 1 printer dot (ensured by setting the line width to 0) print much faster than wide lines. The differences between line widths and styles in print speed become more apparent when printing data sets of 100,000 points or more.

See <u>Known Problems</u> for a description of printing problems along with possible solutions. Problems not referenced in that topic should be referred to the author as described in the <u>Technical Support</u> topic.

## 烤 Related macro commands

**FilePrint** 

## <u>FilePrintMultiple</u>

See also: <u>Fonts</u> <u>Decimate</u> <u>Symbol/Line Styles</u> <u>Colors</u>

## 4.1.10 Macro command

Displays the Macro dialog box

#### Macro Name

Type or select the macro you want to run or edit.

# Run

Runs the selected macro. The Run button is disabled if an existing macro is not selected.

## Edit/Create

Opens a <u>macro-editing window</u> for a new macro whose name you type in the *Macro Name* box. If you select or type the name of an existing macro in the *Macro Name* box, clicking this button displays the contents of the selected macro in a macro-editing window for review or editing. This button is disabled if the *Macro Name* box is empty.

## Delete

Deletes the selected macro. This button is disabled if an existing macro is not selected.

#### Import

Allows you to import a macro created on another system. The imported macro file will be copied to the / Macros folder below the Application Data\DPlot folder.

#### Cancel

Closes the dialog box.

#### 烤 Related macro commands

FileMacro SaveUndo

### See also:

Sending data to DPlot from another application Creating and/or running a macro

### 4.1.10.1 Creating and/or running a macro

A macro is a series of DPlot commands that you can group together to automate certain tasks. Macros are especially useful in performing common operations on a group of files. The macro editing window is shown below:

🔀 DPLOT Macro Editor
<u>F</u> ile <u>E</u> dit
, 🖘 🔒 😸 🔏 🗎 🗂 🗥 🥹
Courier New 🔹 10 👻
RedToYellow
SaveUndo() ' ' This macro changes the Color scheme used ' by DPlot to vary from red for curve #1 to
' yellow for curve #16.
' active plots will be redrawn 16 times!
PostponeRedraw() Color(1,255,0,0) Color(2,255,17,0) Color(3,255,34,0)
Color (4, 255, 51, 0) Color (5, 255, 68, 0) Color (6, 255, 85, 0)
Color (7,255,102,0) Color (7,255,119,0)
Color (9,255,136,0) Color (10,255,153,0)
Shortcut: none

The name of the macro currently displayed in the text box is shown at the top of the window.

The text box shows a series of macro commands. Macro commands may be preceded by one or more spaces or tabs. (To enter a tab in the text box, press CTRL+TAB). Each line may contain only one macro command.

#### **Comment lines**

Comments may be included in a macro by preceding the text with an apostrophe. DPlot ignores comments when the macro is run. If the first non-space, non-tab character in a line is an apostrophe, or if the first 3 characters are "REM", the entire line is ignored. End-of-line comments are allowed using a leading apostrophe. Blank lines or lines consisting entirely of spaces and/or tabs are ignored.

## Toolbar

⊊≣

- Select the Run button to send the macro commands to DPlot.
- Select the Step button to single-step through a macro, pausing after each command is processed. This feature might be useful for debugging a macro.
- Select the Stop button to terminate macro processing. Note: Most macros are completed in such a short time that the Stop button will usually be ineffective unless you are single-stepping through a macro with the Step button.
  - The Save button is used to save the contents of the macro to a file, and to update the contents of DPlot's settings file so that it will be aware of the location and filename for the macro. Filenames

are automatically generated by the macro editor, and the macro files are placed in the \MACROS folder below the Application Data\DPlot folder.

It is **not** necessary to save a macro in order to run it from the Macro editor. However, if you want to use the macro in the future, you must save it.

If you make changes to a macro, the editor will prompt you to save it before opening a new macro or exiting the macro editor.

Select the Print button to print the contents of the text box to the default printer (set with Control Panel).



5

Select the Cut button to remove selected text and put it on the Clipboard.

Select the Copy button to copy the selected text to the Clipboard.

The Paste button inserts a copy of the Clipboard contents at the insertion point, replacing the selection (if any) with the text on the Clipboard

A

Select the Find button to search the text box for a word or phrase.

Select the Help button to display Help for the macro command found at the insertion point. If the insertion point is not at a macro command, selecting the Help button (or pressing F1) brings up this Help topic.

The Shortcut list box allows you to select a keystroke combination which will automatically run the macro from within DPlot. Before saving the macro, the editor checks the selected shortcut against any existing shortcuts for other macros, and displays an error message if a conflict exists.

To copy all or a portion of one macro to another:

- Select the text of the source macro, either by pressing the Shift key in combination with the arrow keys, or by dragging the mouse over the text you want to copy.
- Press Ctrl+C or click on the Copy button.
- Switch to DPlot and select the Macro command from the File menu. Enter a name for a new macro or select an existing macro to edit.
- Click on the Edit/Create button, launching the Macro Editor.
- In the Macro Editor window, move the cursor to the desired location, then press Ctrl+V or click on the Paste button.

The mechanism used to run macro commands is very similar in function to the methods outlined in Sending data to <u>DPlot from another application</u>. Both methods use Dynamic Data Exchange (DDE) to send commands to DPlot. Macros have the following advantages over the "another application" method:

- Macros do not require the user to possess a programming language that supports DDE (such as Visual Basic) or a third party application that allows DDE commands (e.g. Microsoft Word or Microsoft Excel).
- Macros can be activated by shortcut keys.
- Online Help is more readily available for macros.

However, the Visual Basic environment, as well as that of other DDE-aware applications, may provide you with features that are not possible to implement in macros. For example, it is relatively easy to build a Visual Basic application which generates data points and to then send those data values to DPlot.

For a list of all available macro commands, see Executing DPlot commands from another application.

## 4.1.11 Resize Arrays command

Allows you to change the maximum number of curves that may be plotted, and the maximum number of data points per curve. In general, DPlot will automatically increase the size of the arrays when needed. However, DPlot will run more efficiently if the array sizes are selected such that they will not need to be increased.

The maximum number of curves may not be greater than 100, but the maximum number of points per curve is limited only by the amount of free memory present on your machine.

Values entered with the Resize Arrays command will be used for the initial array sizes for any new document windows that are subsequently opened.

## Shortcut:



Double click the "Max Size:" entry on the status bar.

## Related macro commands

**FileArrays** 

## 4.1.12 1, 2, 3, 4 command

Use the numbers and names listed at the bottom of the File menu to open one of the documents you recently worked on (DPlot File Format). Select the number or name that corresponds to the document you want to open.

See also: Open command

## 4.1.13 Exit command

This menu option causes DPlot to close all open documents, save default parameters, and exit.

Shortcut:

ALT+F4

### 📁 Related macro commands

**FileExit** 

# 4.2 Edit Menu

#### Undo...

Allows you to undo the last 100 operations. If you find that performance suffers when working with large data sets and you can live without this feature, disable it under <u>Options>General</u>. **Note:** Individual <u>macro commands</u> cannot be undone. To preserve the state of a plot before running a macro, the macro should use the <u>SaveUndo</u> command.

Shortcut:



#### Redo...

Reverses the operation of the last Undo.

## Shortcut:

Click on the toolbar.

### **Copy Picture**

Copies a metafile and a bitmap image to the Clipboard.

#### Copy Data Values (XY plots only)

Copies the data points of a single curve to the Clipboard.

## **Copy Dimensions**

Allows you to change the size of bitmaps and/or metafile pictures that are copied to the Clipboard by the Copy Picture command.

#### Paste

Creates a new curve using data from the Clipboard.

## **Paste Special**

Opens a dialog box from which you can choose to establish a link with the source of the data, specify the data type (1D groups, X & Y values, X,Y,Z values for a 3D surface or scatter plot, or X,Y,Z,amplitude values for a 4D surface or scatter plot). You can also use this command to tell DPlot that the clipboard text contains a column of point labels.

## **Insert Link**

Creates a new curve from data in another application, and links the curve to the source of the data.

NOTE: All Edit menu commands below are disabled if the currently active document contains linked data (added with Insert Link).

#### Move/Copy Curves (XY plots) or Move/Copy Groups (1D plots)

Allows you to easily move or copy one or more curves/groups from the currently active document to another open document or a new document.

#### 烤 Related macro commands

**MoveCurves** 

#### Edit Data

Edit data values for a new or existing curve.

#### Crop

Delete all data points that lie outside a rectangle you define with the mouse. XY plots and 3D/4D surface plots only. This command is disabled for 3D views of surface plots.

#### XY plots only

#### **Combine curves**

Allows you to append the data values from one curve to another curve. The curve with the source data will be deleted. You may optionally sort the resulting curve in ascending X order.

## 烤 Related macro commands

CombineCurves

#### **Operate on X**

Allows you to manipulate the X coordinates using an equation.

## *s* Related macro commands

EditAddToX EditMultiplyX EditOperateX

### **Operate on Y**

Allows you to manipulate the Y coordinates using an equation.

## Related macro commands

EditAddToY EditMultiplyY EditOperateY

#### **Decimate Points**

Provides three different methods for reducing data.

#### Delete Points in Box Use your mouse cursor to define a box inside which all data points will be deleted.

# Equal Intervals

Generates a new curve with a constant spacing on the X axis.

#### Erase Curve

Wipes out all data associated with one or more curves.

Shortcut:

Click on the toolbar.

Related macro commands

**EditErase** 

## **Filter**

Eliminates noise or otherwise alters a time history record.

#### **Remove Random Spikes**

Removes individual data points that depart from the trend of a curve by a user-specified criteria.

#### **Remove Trend**

Removes any trend from the amplitude by subtracting a straight line through the first and last points (i.e. make the first and last amplitudes equal to zero).

## 烤 Related macro commands

EditRemoveTrend

#### **Replace Data**

Replace the data for one curve with the data from another curve, and then delete the source curve. This command is particularly useful when updating an existing saved plot with new data.

## 烤 Related macro commands

#### EditReplaceData

#### Normalize

Adjusts the amplitudes of selected curves so that they range from 0-1 or 0-100: Y=scale*(Y-Ymin)/(Ymax-Ymin)

## Related macro commands

EditNormalize

## **Smooth**

Smooths a data record using a low-pass filter.

#### Sort

Sorts the data for a selected curve in ascending abscissa (X) order, then in ascending Y order for equal X values. Many DPlot operations are optimized for monotonically increasing values of X, while others may **require** sorted X values.

Related macro commands

EditSort

#### Swap X,Y

Switches the X and Y values for one or more curves. If the new curve has progressively smaller X values, then the order is automatically reversed so that X increases.

### 💋 Related macro commands

**EditSwapXY** 

### Truncate X

Eliminates all data with independent variable outside a user-specified range. If no data remains, the curve and all associated data are erased.

#### 烤 Related macro commands

### EditTruncate

### Clip Y

Eliminates all data points with amplitudes outside a user-specified range. If no data remains, the curve and all associated data are erased. If instead of deleting points you want to simply ignore points outside a range, see the <u>Amplitude Limits</u> command on the Options menu.

#### Related macro commands

## <u>ClipY</u>

#### YYYYMMDDHHMM to Serial

This command translates X values that are dates in the form YYYYMMDDHHMM (for example January 19, 2003 5:25 PM would be 200301191725) to the Excel 1900 date system. This command also automatically turns on Date and Time number format for the X axis and adjusts the time format setting dependent on the granularity of the input values.

#### Data Processing mode only

#### **Baseline Shift**

Performs the same function as Add a constant to Y, but calculates the desired shift and automatically regenerates integrated data in "data processing" mode.

#### List Editions

Displays a list of all baseline shifts done on a record in "data processing" mode.

#### **Undo Shift**

Reverses the previous Baseline Shift performed in "data processing" mode.

## 3D plots only

#### Operate on Z

Allows you to manipulate the Z coordinates using an equation.

### 烤 Related macro commands

### EditOperateZ

#### Swap X,Y

Switches the X and Y values for all data points. This option might be useful if, for example, your 3D input files are ordered latitude, longitude, elevation (rather than longitude, latitude, elevation).

## **Number Triangles**

Numbers triangles on a 2D contour plot of randomly-spaced 3D points. This option is useful when deleting individual triangles using the right-click context menu.

#### **Delete Triangles**

Allows you to delete one or more triangles from a 2D contour plot of randomly-spaced 3D points. You might find this option useful in either creating holes in the surface of the plot or in overriding DPlot's creation of a convex mesh. For more information see the How To topic "How do I force DPlot to create a triangular mesh of my 3D points that is not convex?", and for an alternative solution see the Define Boundary command on the Options menu.

## **Delete Thin Triangles**

Removes triangles with interior angles less than 5 degrees if the associated triangle vertex is on a free edge. Thin triangles often do a poor job of mapping a surface. Triangle removal is recursive: if triangle B has an interior angle less than 5 degrees which borders triangle A, and triangle A is removed, then triangle B will also be removed.



#### See also

**Data Processing Command** 

#### 4.2.1 **Copy Picture command**

Copies a metafile and a bitmap image to the Clipboard. A metafile preserves all of the information content of a plot. The metafile picture can be scaled and rotated without distortion, and individual graphic entities (fonts, for example) may be edited. A metafile is device-independent, so the resolution of the image will be that of the final destination output device. Metafiles, however, are **not** created for any view of surface plots with filled color bands (simply because the format is not designed for this type of image and would be very large if created). The bitmap image created by DPlot differs from the normal Windows screen capture in that the image size is not necessarily the same size as the document client area. DPlot will instead use the dimensions specified with the Copy Dimensions command. The title bar, menu, and window borders are not included in the image.

#### Shortcut:



## **Output size**

If "Set Plot Size" on the Extents/Intervals/Size dialog box is not checked, DPlot produces a metafile with dimensions specified by the Copy Dimensions command. These dimensions may be changed in the destination application, but you then risk not being able to match font sizes with the intended size. For best results, you should enter the desired dimensions for the entire picture using the Copy Dimensions command, or dimensions for the box surrounding the plot using the Extents/Intervals/Size command before copying the plot to the Clipboard. Dimensions entered with Extents/Intervals/Size take precedence over those entered with the Copy Dimensions command.

## **Clipboard format**

Many applications that allow you to paste data from the Clipboard will accept a variety of clipboard formats. Microsoft Word, for example, will accept either a bitmap or metafile picture. The target application determines which format to use when the Paste command is selected. Most applications that accept multiple clipboard formats have a Paste Special command in addition to the Paste command. To select a format different than the default, use the Paste Special command in the target application.

### Link

In addition to a metafile picture and bitmap DPlot also copies "link" information to the Clipboard. Any Windows application that supports a *Paste Link* command can be linked to the DPlot plot, so that changes made to the plot by DPlot are automatically transferred to the client application. For example, in Microsoft Word select the *Paste Special* command, check the *Paste Link* box, and select *OK*. Any subsequent changes to the plot in DPlot will cause the picture in Word to be automatically updated. Please note that when using the *Paste Link* option it is up to the client application (not DPlot) to select the desired format (metafile or bitmap). Word, for example, will **always** request an updated metafile picture rather than a bitmap, even if "bitmap" is selected in the *Paste Special* dialog.

### Problems with metafiles

Please see the <u>Known Problems</u> section for a description of problems with various Office products, and their solutions.

## Related macro commands

CopyPicture SetBitmapDims SetImageCrop SetMetafileDims

#### See also

Transferring DPlot Plots to Other Applications Copy Data Values

## 4.2.2 Copy Data Values command

Copies the data values of a selected curve to the Clipboard as text. The data may then be imported into a number of Windows applications that support the "Paste" command. To copy the values of multiple curves from DPlot to another Windows application, repeat the Copy Data Values ... Paste sequence for each curve.

You may also "link" DPlot data to any application that has a "Paste Link" command. For example, in Microsoft Word select the "Paste Special" command from the Edit menu, check the "Paste Link" box, and select "OK". Any subsequent changes (in DPlot, not Word) to the data copied to the Clipboard will be automatically reflected in the Word document.

## 烤 Related macro commands

<u>CopyText</u>

See also Copy Picture command

## 4.2.3 Copy Dimensions command

Allows you to specify the sizes of bitmaps and/or metafile pictures that are copied to the Clipboard by the Copy Picture command. When DPlot is first installed, the default bitmap dimensions are 0x0 (which tells DPlot to use the current document client area dimensions), and the default metafile picture dimensions are 6.0x4.5 inches.

NOTE: The metafile picture dimensions are only used when the "Set Plot Size" option of the <u>Extents/</u> <u>Intervals /Size</u> command is unchecked. If checked, the corresponding dimensions are used for all output devices, including metafile pictures. See also Copy Picture command

## 4.2.4 Paste command

Creates a new curve with data from the Clipboard. The source of the data may be a word processor, spreadsheet, or any Windows application capable of copying data to the Clipboard, as well as a DOS application running in a window.

### Shortcut:



### Data organization

For XY plots, data should be organized as X-Y pairs, or X followed by multiple Y values for each X. Each X data value must start on a new line. For 3D plots, each line should consist of X,Y,Z values. Values may be separated by spaces, commas, tabs, or semicolons. If the <u>decimal symbol</u> is a comma, the columns should be separated by tabs or semicolons. For 1D plots, the first line should contain a character string description of the group, followed by the amplitudes for the group with 1 or more values per line. For XY and 3D plots, data may be preceded by one line of column headings. If present, headings will be used for the axis labels or the legend in the case of XY plots with multiple curves.

**Please note:** When you first start DPlot, the blank document window is set up for an XY plot. If you want to paste 1D or 3D data into a new document, either 1) open a new document using the appropriate data type under File>New, or 2) Use the Paste Special command rather than Paste. Otherwise the clipboard contents will be interpreted as 2D data.

#### Formats

For XY plots, in addition to numbers DPlot will also accept columns consisting of dates, times, datetime pairs, currencies, and percentages. Date format is flexible but entries **must** be separated by a dash (-) or forward slash (/). If the month is specified as a number, then DPlot assumes the order is m/d/y **unless** you use 4 digit years. These date forms (for Jun 8, 2005) are acceptable:

6/8/05 06/08/2005 6-8-2005 2005/6/8 8-Jun-05 June-08-2005

but 'June 8, 2005' is not.

Time values should be in the form h:m:s AM/PM, using ':' as the separator. Leading zeroes are acceptable. If the AM or PM designation is omitted, DPlot assumes a 24 hour clock.

Date-time pairs should be in the form 'm/d/yy h:m:s AM/PM'.

Only dollar signs and British pound signs are accepted as currency symbols. The currency symbol should precede the value, as in '\$56.23'. If monetary values include a comma for the thousands separator, the entry **must** be surrounded by double quotation marks. This is the same scheme used by Excel when saving CSV files.

Percentages should be followed by a percent sign (%). DPlot divides the number by 100 and uses Percent number formatting on the associated axis when appropriate.

Any other data type will be ignored, but in general will be allowed. If blank entries are included, the columns **must** be tab-separated or comma-separated to be interpreted correctly. Entries must not

contain commas or tabs unless they are surrounded by double quotation marks.

#### Copying data from a DOS application

You may copy data either from a full-screen application or a windowed application. However, when running a full-screen application, you can copy only an image of the contents of the entire screen onto the Clipboard, so you increase the likelihood of copying invalid characters.

To copy selected information onto the Clipboard

- 1 Click the application's Control-menu box. Or press ALT+SPACEBAR to open the Control menu.
- 2 From the Control menu, choose Edit.
- 3 From the Edit cascading menu, choose Mark.
- 4 In your document, select the information you want to copy.
- 5 Click the application's Control-menu box. Or press ALT+SPACEBAR to open the Control menu.
- 6 From the Control menu, choose Edit.
- 7 From the Edit cascading menu, choose Copy.

### Selated macro commands

Paste

## 4.2.5 Paste Special command

This command allows you to specify whether data on the clipboard is 2D, 3D, etc. You may also elect to link to the source application, if the source provides link information. Finally, you can use this command to tell DPlot that the clipboard text contains a column of point labels.

#### **To create a link between a source document and a dependent plot in DPlot** Both applications must be Microsoft Windows applications.

- 1 Open the source document in the server application.
- 2 Select the source data you want to copy into the dependent document in DPlot. See the documentation for the server application for more information on selecting data.
- 3 From the Edit menu, choose Copy.
- 4 Switch to the dependent document window in DPlot.
- 5 Select Paste Special from the Edit menu. Check the Link box, change the data type if necessary, then click OK.

If the Link box is disabled, the other application does not support linking or cannot supply the data in a format that DPlot can use.

#### Saving links

Link information is normally saved to a DPlot file, so that when you reopen the file the link will be automatically established if the source application is open, or you will be prompted for whether you want to launch that application if it is not open.

Link information **only** saved if the link is still active. Links are deleted when the source application is closed. So if you want, for example, to save a link to an Excel spreadsheet within a DPlot file so that the link is reestablished when you reopen the DPlot file, save the DPlot file **before** you close the Excel spreadsheet.

### Data organization and Format

Data should be organized and formatted as described under the Help topic for the Paste command.

#### 💋 Related macro commands

**PasteLink** 

## **RemoveLink**

## 4.2.6 Insert Link command

Creates one or more new curves or a new 3D surface with data from a DDE server application and establishes a link with the server. When the source data changes, the DPlot plot reflects those changes.

**To create a link between a source document and a dependent plot in DPlot** Both applications must be Microsoft Windows applications.

- 1 Open the source document in the server application.
- 2 Switch to the dependent document window in DPlot.
- 3 Select Insert Link from the Edit menu.
- 4 Type the name of the application in the "Application" box (e.g. Excel, WinWord, or MsAccess).
- 5 Type the name of the DDE topic in the "Topic" box. For most DDE servers, the topic is the name of the source datafile. If the file has not been saved to disk, the topic name may be the caption of the window containing the data (e.g. Sheet1 in Microsoft Excel or Document1 in Microsoft Word). For Microsoft Access, the topic name will typically be a database name plus a table name, separated by a semicolon, as in 'mydata.mdb;TABLE Table1'
- 6 Type the name of the data item you want to link to in the "Item" box. Check the documentation of the server application for valid item names. In Microsoft Excel, items should be cell references in R1C1 format (e.g. R1C1:R100C2 refers to the first 2 columns of 100 rows). Please note: A1 style cell references will not work correctly.

For best results with Microsoft Word the linked item should be tagged with a bookmark, and the item is the name of the bookmark.

Microsoft Access supports a wide variety of options for the DDE item. To retrieve the entire contents of a table including field names, use Item=**all**. To retrieve all table data without field names, use Item=**data**. **Please note:** Unlike Excel and Word, changes to data in Access do not normally cause immediate updates to links. The refresh interval for updating links is controlled within Access on the Advanced tab of the Tools>Options dialog in Access 2002. Other versions may behave differently; check your Access documentation.

#### **Recently used links**

This box shows up to 20 of the most recently used links, in the form Application|Topic!Item. To re-use one of these links, select it from this box and click the *Copy* button.

#### **Current links**

Shows currently active data links for the active document. To remove a link, select it and click the *Remove* button. This action does **not** erase the associated data, it simply removes the association with the server application so that changes to the data in the server will not be reflected in DPlot. Links are **not** saved in undo files for a variety of reasons; if you remove a link in this manner it cannot be restored with the Edit>Undo command.

#### Class entry

This entry is generally not important if the server application is already running. It is only used to find the path to the application within the registry if the server application is not running. When you link data to DPlot using the Link option of the <u>Paste Special</u> command, the Class value is retrieved from the Clipboard ObjectLink data when available (e.g. Excel=Excel.Sheet.8, WinWord=Word.Document.8).

#### Saving links

Link information is normally saved to a DPlot file, so that when you reopen the file the link will be automatically established if the source application is open, or you will be prompted for whether you want to launch that application if it is not open.

Link information only saved if the link is still active. Links are deleted when the source application is

closed. So if you want, for example, to save a link to an Excel spreadsheet within a DPlot file so that the link is reestablished when you reopen the DPlot file, save the DPlot file **before** you close the Excel spreadsheet.

#### **Data organization and Format**

Data should be organized and formatted as described under the Help topic for the Paste command.

NOTE: All Edit menu commands below Insert Link are disabled if the currently active document contains linked data.

#### 💋 Related macro commands

InsertLink RemoveLink

## 4.2.7 Edit Data command

Allows you to edit data for an existing curve, surface, or group or enter data values for a new curve, surface, or group. The maximum number of points that may be edited using this command under Windows 95 or 98 is 32767. For newer versions of Windows there is no limit.

## Shortcut:

Click on the toolbar.

#### OK

Accepts the input and closes the dialog box.

For XY plots you of course need at least 2 points to draw a line segment, though you can draw a symbol at a single point.

At least 3 non-collinear points (projected onto the XY plane) with at least 2 points having a variation in Z are required for a 3D surface plot. DPlot will remove points as needed to eliminate pairs of points with identical X,Y values, preserving the point with the maximum Z value.

3D scatter plots require at least 2 points with a variation in Z. Unlike surface plots, you **can** have multiple points with identical X,Y values.

1D plots require at least 2 amplitudes per group. Groups with less than 2 points are removed.

## Cancel

Restores previous values and closes the dialog box.

#### Add

Adds a data point to the end of the array.

## Insert

Inserts a data point before the point currently highlighted in the list box.

## Delete

Deletes the data point currently highlighted in the list box.

## Paste Z

(3D data on a rectangular grid). This feature allows you to replace all of a plot's existing Z values with clipboard data, which must be formatted as one value per text line. X and Y will not be modified. This feature might be handy if you frequently produce 3D surface or scatter plots that always have the same X and Y coordinates, but different Z values.

### Replace

(Random 3D data). This feature allows you to replace all of a plot's existing X, Y, and Z values with

clipboard data, which must be formatted as three values per text line. This feature might be handy if you frequently produce 3D surface or scatter plots with a complicated setup: embedded images, long notes, etc.

## XY Plots Only

You can edit points and/or add a point by clicking on the plot with your mouse. To edit the currently selected point, click the left mouse button at the desired location. To add a new point, press the Ctrl key while clicking. If the *Snap mouse* box is checked, the coordinates of the point selected with the mouse will be rounded to "nicer" values: one tenth the tick mark interval on linear scales, for example. If you want to enter a new point that lies outside the extents of the graph, use the *Zoom out* button or one of the 4 pan buttons. The zoom and pan buttons may be disabled, dependent on the scaling type. For example, you cannot pan with a polar plot.

### **3D Plots**

For 3D data on a rectangular grid, a Z value less than -1.E+37 will be ignored. In this way you can create a plot on a rectangular grid that does not have a rectangular boundary, and/or create holes in the surface.

## 4.2.8 Decimate command

Provides three different methods for reducing the number of data points in a curve. Since all of the decimation processes are irreversible, DPlot gives you the option of either replacing or preserving the curve that is operated on. Select the **Replace existing curve** checkbox to replace the original curve with the decimated data. Leave the box unchecked to preserve the original data and produce a new, decimated curve.

### **Average Points**

Averages a user-specified number of amplitudes. Peak values may be lost.

## Welch Depopulation Scheme

Preserves only those points that prevent the output record from departing from the original data by more than a user-specified ratio of the range between minimum and maximum input values. Peak value is preserved. Generally has the best performance of the three methods, but may lose detail in low-amplitude regions of the data. Developed by Dr. Bob Welch of the US Army Engineer Research and Development Center in Vicksburg, Mississippi.



### **Skip Points**

Preserves every Nth point, where N is user-specified. Peak values may be lost.

### 📁 Related macro commands

**EditDecimate** 

See also Resize Arrays

## 4.2.9 Filter data command

The Filter command provides several methods for digitally filtering time histories. All of the filters employed by DPlot use the following linear filter:

$$\boldsymbol{y}_n \ = \ \sum_{k=0}^M \, \boldsymbol{c}_k \ \ \boldsymbol{x}_{n-k} \ + \ \sum_{j=1}^N \, \boldsymbol{d}_j \, \boldsymbol{y}_{n-j}$$

where x = input amplitudes y = output amplitudes

Dialog box options for the various filters are described below:

#### Filter type

**User defined.** Values for the C and D arrays are set by the user. This option is not recommended for users unfamiliar with signal processing.

**Frequency rejection (Carleton).** Removes the frequency (specified in the "Reject frequency" box) from a time series using the methods described in WES Miscellaneous Paper N-70-1, "Digital Filters for Routine Data Reduction", by H.D. Carleton.

**Frequency rejection (Press).** Removes a narrow frequency band from a time series using the procedures described in "Numerical Recipes", Cambridge University Press, by W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. The frequency to remove is specified in the "Reject

frequency" box. The width of the frequency band to remove is specified in the Bandwidth box.

**Bandpass.** Preserves the frequencies between the lower and upper cutoff frequencies, following procedures described in "Numerical Recipes" (see above). For a low-pass filter, set the lower cutoff frequency to 0 and the upper cutoff value to the desired limit; for a high-pass filter do the opposite: set the upper cutoff frequency to a very high frequency and the lower cutoff to the desired value. Note that this filter does not have a very sharp cutoff. To obtain a sharper cutoff you can apply this filter multiple times. **NOTE:** For an alternative, effective low-pass filter see the Help topic for the <u>Smooth</u> command.

Zero frequency rejection. Removes an offset from an oscillating time function.

**Option** *x* **low-pass.** The four low-pass filters are described in the <u>Low Pass Filter</u> Help Topic. These filters each use the linear filter equation described above, but make multiple passes using different filter coefficients.

**CFCx.** Uses the above digital filter for a <u>CFC</u> (Channel Frequency Class) value of 60, 180, 600, or 1000 according to <u>SAE J211</u>. The minimum sampling frequency for each of these filters is 10 times the CFC number, in Hertz. For example the minimum sampling frequency for a CFC60 filter is 600 Hz (timestep no greater than 0.016667 seconds), while for a CFC1000 filter is 10 kHz (timestep no greater than 0.1 milliseconds). Filter coefficients are determined by:

$$\omega_d = 2\pi \left(\frac{C}{0.6}\right) \bullet 1.25$$
$$\omega_a = \tan \left(\frac{\omega_d T}{2}\right)$$
$$C_0 = \frac{\omega_a^2}{1 + \sqrt{2}\omega_a + \omega_a^2}$$
$$C_1 = 2C_0$$
$$C_2 = C_0$$
$$D_1 = \frac{2(\omega_a^2 - 1)}{1 + \sqrt{2}\omega_a + \omega_a^2}$$
$$D_2 = \frac{-1 + \sqrt{2}\omega_a - \omega_a^2}{1 + \sqrt{2}\omega_a + \omega_a^2}$$

where C = CFC number, T = timestep in seconds.

#### Time scale

The time scale is important for each of the **Frequency rejection** and **Bandpass** filters and the **CFC** filters, since the coefficients C and D are dependent on the sampling frequency of the input.

#### Passes

There are two considerations with this entry: 1) Each of the filtering methods used by DPlot introduce a small phase shift to the input record. This situation may be corrected by entering a value of 2 in the **Passes** box. This will repeat the filtering operation in time-reversed order. 2) In some cases, particularly for noisy records, it may be desirable to repeat the operation more than twice. Odd values for **passes** always end in a forward pass; even values end in a reverse pass.

#### **Replace data**

Check this box to replace the existing record with the filtered data. Leave this box unchecked to produce a new curve. This option is disabled in data processing mode (the existing record is always replaced).

#### **Filter coefficients**

These controls are disabled for all but the **User defined** filter. The coefficients used by DPlot for each of the other filters other than **Option** *x* **low pass** (frequency rejection, zero frequency rejection, CFC*x*) are shown but cannot be edited directly.

烤 Related macro commands

EditFlter

## 4.2.9.1 Low Pass Filters

The low pass filters used by DPlot are composites of several filters using the linear equation shown in the <u>Filter</u> topic. This technique was first suggested by J.R.B. Whittlesey (see below).

A full description of filtering techniques is well beyond the scope of this documentation. For more information see the references at the end of this topic.

Four composite low-pass filters are constructed by combining the following components:

Operator	to be Performed to	Function
Designation	Yield Each Displacement Output	(z Plane)
S1	$y_n = X_n + X_{n-1}$	F(z)=1+ z
S2	$Y_n = X_n + X_{n-2}$	$F(z) = 1 + z^2$
S3	$y_n = X_n + X_{n-3}$	$F(z) = 1 + z^3$
S4	$y_n = X_n + X_{n-4}$	$F(z) = 1 + z^4$
S6	$y_n = X_n + X_{n-6}$	$F(z) = 1 + z^{6}$
S8	$y_n = X_n + X_{n-8}$	$F(z) = 1 + z^8$
SX	$Y_{n} = X_{n} + X_{n-1} + X_{n-2}$	$F(z) = 1 + z + z^2$
R3	$y_{n} = 0.43(x_{n} - y_{n-3}) + x_{n}$	$F(z) = \frac{1.43}{2}$
		$1 + 0.43z^3$
R6	$y_{n} = 0.36 (x_{n} - y_{n-8}) + x_{n}$	$F(z) = \frac{1.36}{1.36}$
		1+0.36z ⁶
R9	$y_n = 0.32(x_n - y_{n-9}) + x_n$	$F(z) = \frac{1.32}{2}$
	,	1+0.32z°
R12	$y_n = 0.324 (x_n - y_{n-12}) + x_n$	$F(z) = \frac{1.324}{2}$
		$1+0.324 z^{12}$

In the above table x represents the input amplitudes, y represents the output.

Each of the 4 low-pass filter options applies several operators in sequence. The output from one operator is the input to the second operator. Each of the operators shown above amplifies the output, so that the original input is effectively multiplied by some factor. To compensate for this effect, DPlot multiplies the final output by the gain factors shown in the table below.

Low Pass Option	Sequence	Factor
1	S1,SX,S2,R3	1/12
2	S1,SX,S2,S2,S4,R6	1/48
3	S1,SX,S3,S3,S6,R9	1/48
4	S1,SX,S2,S4,S4,S8,	1/96
	R12	

For more information see:

Blackman, R.B. and Tukey, J.W., "The Measurement of Power Spectra, for the Point of View of Communications Engineering," **The Bell System Technical Journal**, Vol 37, January and March 1958,

pp 185-288, 485-569.

and

Whittlesey, J.R.B., "A Rapid Method of Digital Filtering," **Communications of the Association for Computing Machinery,** Vol 7, No. 9, September 1964, pp 552-556.

Note: The Option 3 and 4 filter sequence is modified slightly from the above references to give an amplitude response of 1.0. The original filter designs might have exceeded 1.0. (That is, the output amplitude might have been greater than the input.)

## 4.2.10 Baseline Shift command

Allows you to specify a new value at a specific time in a record generated from the primary data file in "data processing" mode.

If the **Constant Baseline Shift** checkbox is checked, then the shift is applied to <u>all</u> points in the primary record. Otherwise, you may specify a start time and end time.

The start time **must** be less than the time at which you want the integrated record to match the value you specify.

After altering the primary record, DPlot will automatically regenerate integrated records and, if the shift is applied to <u>all</u> points, will update the **CBS** display in the title.

DPlot maintains a list of all shifts applied to the primary record, which can be displayed with the List Editions command.

See also

**Data Processing** 

## 4.2.11 List Editions command

Displays a list of baseline shifts made to the primary data record in "data processing" mode. Includes shifts made with either the "Add a constant to Y" or "Baseline Shift" commands.

See also Edit Menu Data Processing

## 4.2.12 Smooth command

Smooths the amplitudes for a selected curve.

The X values for the selected curve(s) must be monotonically increasing. If no curves possess that characteristic, this command is disabled.

NOTE: If your data is not sorted and you are certain it is appropriate to sort it, you can use the Sort command on the Edit menu to sort your data points before using this command. Sorting isn't appropriate for data sets describing measurements that, by design, have two or more Y values for any unique X, e.g. moment-thrust diagrams.

The Smooth command removes any linear trend to the data so that the first and last amplitudes are zero, uses a Fast Fourier Transform to low-pass filter the data, and finally reinserts the linear trend to the amplitudes.

The "smoothing window" is the number of points over which the data should be smoothed. Zero gives no smoothing at all, while a value approaching half the number of data points will render the data virtually featureless.

The algorithm used by the Smooth command is taken from "Numerical Recipes" by Press, Flannery, Teukolsky, and Vetterling. This reference includes the following qualification:

Data smoothing is probably most justified when it is used simply as a graphical technique, to guide the eye through a forest of data points all with large error bars. In this case, the individual points and their error bars should be plotted on the same graph, and no quantitative claims should be made on the basis of the smoothed curve. Data smoothing is least justified when it is used subjectively to massage the data this way and that, until some feature in the smoothed curve emerges and is pounced on in support of an hypothesis.

As an alternative to the <u>Filter</u> command with the *Bandpass* option, you can low-pass filter your data given the correct "smoothing window". Use a smoothing window of 1/(cutoff frequency * time increment). For example, for a 60Hz low-pass filter on data with a time increment of 1 millisecond, use 1/(60 cycles/sec * 0.001 sec/point) = 16.6667 points.

## Related macro commands

**EditSmooth** 

See also Fast Fourier Transform Filter

## 4.2.13 Equal Intervals command

Generates a new curve with a constant spacing on the abscissa (X axis). This command is useful for generating evenly spaced data points prior to performing some function that requires this characteristic (FFT for example). It may also be used to generate a smooth curve through widely spaced data points.

This command uses one of four methods for interpolating amplitudes between existing data points. The Linear, Quadratic, and Cubic interpolation methods use, respectively, 1st, 2nd, and 3rd order polynomials.

If the input curve is not sorted in increasing X order, a sorted copy of the input is generated and operated on. Similarly, points with duplicate X values are removed from the input, with the Y value at the point set to the average of all points sharing that same X. Neither of these operations alter the input data.

■ Linear example:





Polynomial interpolation uses a finite number (2,3, or 4) of "nearest neighbor" data points to give interpolated values of the input record. In general, the interpolated values do not have continuous first or higher derivatives. That happens because, as X crosses the tabulated input values, the interpolation scheme switches which input data points are used to derive the polynomial interpolating function. If continuous first and second derivatives are important the Cubic Spline method should be used. The cubic spline implemented by DPlot sets the second derivatives at each end of the input record to zero.

□ Cubic Spline example:



Select the **Replace existing curve** checkbox to replace the original curve with the evenly spaced data. Leave the box unchecked to preserve the original data and produce a new curve.

If **Use start, end X values** is unchecked, the X values for the new curve will range between the minimum and maximum X values for each selected curve. If **Use start, end X values** is checked, then the specified X limits will be used for all selected curves.



# 4.2.14 Operate on X command

Allows you to manipulate the abscissa (X coordinates) of an XY plot using an equation that is a function of the current X values, Y values, and/or the 1-based point index J.

Available operators are +, -, *, /, ^,>,<, and =. The inequality operators (>, <, and =) result in 1.0 if the condition is true, 0.0 otherwise.

Function (sin, cos, etc.) arguments must be in parentheses. See the list of available functions.

You can use placeholder codes for certain data parameters. For example "\$YMAX(2)" will be replaced with the maximum Y value of the second curve. For more information on placeholder codes click <u>here</u>.

You can create conditional equations with the **if** function. The **if** function takes the form if(*test*, *if_true*, *if_false*), where *test* is the condition you want to check, *if_true* is the value the equation takes if *test* is true, and *if_false* is the value taken if *test* is false. For example, "if x > 5 then  $y=x^2$ , otherwise  $y=(10-x)^2$ " can be expressed with **if**( $x<5,x^2,(10-x)^2$ ).

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute the value of pi (3.141592653589793...) for "PI", "pi", or "Pi", and the 1-based point index for "J" or "j". For example, 0.25*J will produce evenly-spaced X values starting at X=0.25.

NOTE: Be specific. DPlot does not interpret "2X" as "2*X".

Examples:

X=

X+2 Shifts all X values to the right by 2.

X*2 Multiplies all X values by 2.

log10(X) Sets X values to the base 10 logarithm of their current values.

## Related macro commands

**EditOperateX** 

## 4.2.15 Operate on Y command

Allows you to manipulate the amplitudes (Y coordinates) of an XY plot using an equation that is a function of the current X values, Y values, and/or the 1-based point index J.

Available operators are +, -, *, /, ^,>,<, and =. The inequality operators (>, <, and =) result in 1.0 if the condition is true, 0.0 otherwise.

Function (sin, cos, etc.) arguments must be in parentheses. See the list of available functions.

You can use placeholder codes for certain data parameters. For example "\$YMAX(2)" will be replaced with the maximum Y value of the second curve. For more information on placeholder codes click here.

You can create conditional equations with the **if** function. The **if** function takes the form if(*test*, *if_true*, *if_false*), where *test* is the condition you want to check, *if_true* is the value the equation takes if *test* is true, and *if_false* is the value taken if *test* is false. For example, "if x > 5 then  $y=x^2$ , otherwise  $y=(10-x)^2$ " can be expressed with **if**( $x<5,x^2,(10-x)^2$ ).

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute the value of pi (3.141592653589793...) for "PI", "pi", or "Pi", and the 1-based point index for "J" or "j".

NOTE: Be specific. DPlot does not interpret "2Y" as "2*Y".

Examples:

Y=

Y+2 Shifts all Y values up by 2.
Y^2 Squares all amplitudes.
log10(Y) Sets Y values to the base 10 logarithm of their current values.
Y*1.8 Transforms Y in degrees Celsius to degrees Fahrenheit
+32

📁 Related macro commands

**EditOperateY** 

## 4.2.16 Operate on Z command

Allows you to manipulate the Z coordinates of a 3D data set using an equation that is a function of Z (and/or X or Y).

Available operators are +, -, *, /,  , , -, -, and =. The inequality operators (>, -, and =) result in 1.0 if the condition is true, 0.0 otherwise.

Function (sin, cos, etc.) arguments must be in parentheses. See the list of available functions.

You can use placeholder codes for certain data parameters. For example "\$YMAX(2)" will be replaced with the maximum Y value of the second curve. For more information on placeholder codes click <u>here</u>.

You can create conditional equations with the **if** function. The **if** function takes the form if(*test*, *if_true*, *if_false*), where *test* is the condition you want to check, *if_true* is the value the equation takes if *test* is true, and *if_false* is the value taken if *test* is false. For example, "if x > 5 then  $y=x^2$ , otherwise  $y=(10-x)^2$ " can be expressed with **if**( $x<5,x^2,(10-x)^2$ ).

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute the value of pi (3.141592653589793...) for "PI", "pi", or "Pi"

**NOTE: Be specific.** DPlot does **not** interpret "2Z" as "2*Z".

Examples:

Z=

Z+2 Shifts all Z values up by 2.

Z*2 Multiplies all Z values by 2.

log10(Z) Sets Z values to the base 10 logarithm of their current values.

#### Related macro commands

**EditOperateZ** 

## 4.2.17 Edit Mode command

Allows you to quickly edit specific data points and/or add labels to data points with a point-and-click.

Shortcut: Click in on the toolbar.

This command is disabled for 1D statistics plots and 3D plots of 3D or 4D data. For XY plots, you cannot use this feature to edit generated records in <u>Data Processing</u> mode (for example, velocity curve generated from an acceleration record).

### XY plots

DPlot scans through each curve to find two points that bound the cursor in the X direction. If/when found, it then checks the distance from the cursor hotspot to the line segment defined by those 2 points. A "hit" is registered if this distance is less than 3 pixels. DPlot repeats this operation for all curves and chooses the curve with the smallest distance from the cursor, then picks the closest endpoint of the line segment to edit. For curves with a large number of points (such that the spacing

between adjacent points is less than 1 pixel in the X direction), you will almost certainly need to zoom in on the curve in order to pick a specific point.

You may also use this feature to label points. The label may later be edited with the <u>Label Points</u> command on the Text menu.

For XY plots you can change the values of the X and/or Y coordinate by clicking the mouse on the graph at the desired new location, in addition to entering values in the corresponding text boxes. If the *Snap mouse* box is checked then the coordinates will be rounded to "nicer" values: one tenth the tick mark interval on linear scales, for example.

Next selects the next point in the curve.

Prev selects the previous point in the curve.

**Insert** inserts a data point between the currently selected point and the next point in the curve. The initial X and Y values of the inserted point will be the average X/Y of the current and the next point.

Delete deletes the currently selected data point.

#### Distance to another point

You can right-click the mouse on another data point on the plot and this dialog will report the distance from the current point to that point. **Please note:** the distance reported is simply the square root of the sum of the squares of the horizontal and vertical offsets, in whatever units are used for X and Y. The reported distance may therefore be meaningless. The Y offset may also be in mixed units, depending on whether multiple scales are used on the plot.

### 2D plots of random 3D points

DPlot finds the triangle that you are pointing to, then selects the triangle vertex closest to the cursor hotspot. You may edit the X, Y, and or Z coordinates of this point. If you change either the X or Y coordinate then DPlot will regenerate a triangular mesh after selecting "OK".

#### 2D plots of gridded 3D data

DPlot finds the grid square containing the cursor, and picks the vertex closest to the cursor hotspot. You may then edit the Z coordinate of that spot, but not the X or Y.

## Label this point

Check this box to add a text label adjacent to the data point. **NOTE:** Point labels are not drawn in 3D views of 3D data, only in 2D views.

The control codes \$X, \$Y, and \$Z in the label text will be replaced with the corresponding X, Y, and/or Z components of the labeled point. If the value of those coordinates is later changed, the corresponding field in the label will be updated automatically.

The label will be drawn on the plot after the label control loses the focus (by <TAB>ing away from the text box or clicking another control). Do **not** try to drag the note to a different location while this dialog is active; clicking the mouse on the plot (whether on the label or not) will shift the associated data point to the mouse location.

After dismissing the Edit Mode dialog you can move a label by dragging it with your mouse. You can also change several label attributes by right-clicking on the label and selecting options from the dropdown menu. For more control over the label appearance double-click the label, or select the <u>Label</u> Points command on the Text menu.

## 4.2.18 Remove Random Spikes command

This procedure is useful for removing single samples with high amplitudes which bear no relation to the true record. The default conditions defining a spike are:

$$0.75 < \frac{y_n - y_{n-1}}{y_n - y_{n+1}} < 1.25$$

The above inequality states that the ratio of the two sides of a spike must be nearly unity. You may elect to modify the 0.75 and 1.25 constants, though this is rarely necessary for true spikes.

The condition which defines the sizes of spikes to be removed is:

$$\left|y_n - \left(\frac{y_{n-1} + y_{n+1}}{2}\right)\right| > 0.1 \times \text{Range}$$

where Range is the current range in amplitudes for the record. Again, you can modify the constant (default=0.1) in the above inequality. If any sample in a record meets both of the conditions described above, it is considered a spike and replaced with the average of the adjacent values.

### **Show Spikes**

Click this button to draw small black squares at each data point that meets the spike criteria. If you change any of the above three criteria you can click Show Spikes again to note any change.

## Related macro commands

EditRemoveSpikes

## 4.2.19 Swap X, Y command

Switches the X and Y values for one or more curves on an XY plot or for all values in a 3D surface plot or scatter plot.

For 3D plots this option might be useful if, for example, your input files are ordered latitude, longitude, elevation (rather than longitude, latitude, elevation).

For XY plots, if the graph contains more than one curve you will be prompted for which curves to perform this operation on. If the result has progressively smaller X values, then the order is automatically reversed so that X increases.

For all 3D plots and for XY plots in which this operation is performed on all curves, in addition to swapping the X,Y values this command will swap the values for the following attributes (some of which are only relevant to XY plots):

Options>Extents/Intervals /Size:

User-specified extents Invert X, Invert Y settings User-specified tick mark intervals or number of tick marks per axis. Minor tick mark intervals Plot width and height

Number formats (Options>Number Formats)

Multiple axis settings (Options>Multiple Axes)

Primary and secondary X and Y axis labels. (Text>Title/Axes)

Scaling for XY plots. If the original plot used linear scaling on the X axis and logarithmic scaling on the Y axis, for example, then the resulting plot with use logarithmic scaling on the X axis and linear scaling on the Y axis.

Please note that none of the above modifications are made to XY plots if the graph contains more than one curve and you do not swap X,Y values for all curves.

# 4.3 Text Menu

#### **<u>Title/Axes</u>**

Enter or edit title strings and axis labels.

## Legend/Labels

Enter or edit curve legend, labels, or legend title.

#### Label Points

Add, edit, move, or delete labels for specific data points.

#### Date

Toggle display of the date. The date will be displayed using the "Short Date" format selected with the Control Panel "International" application. The date will be updated to today's date whenever you save the plot as a DPlot file or turn this option on.

## 烤 Related macro commands

ShowDate

### **Print Caption**

Toggle display of the document caption in the lower right corner of the plot. The caption is normally either the filename of the document or the source of the data; e.g. EXCELData or ClipboardData.

## 烤 Related macro commands

**ShowCaption** 

### Add/Edit Note

Enter or edit a notation placed at an arbitrary location on the plot.

#### Force legend/notes on screen

If you use the mouse to drag the plot legend or notes outside the plot limits and then change the plot size with the <u>Extents/Intervals/Size</u> command on the Options menu, it is possible that the legend or notes will now be completely off the display and inaccessible. Select this command to force those items to be visible on the screen. This is **not** a persistent command; it only affects the legend or notes at the time you use it.

## **Fonts**

Change the typeface and/or other attributes for various labels.

## X,Y Labels or X,Y,Z Labels

Specify text labels to appear in place of or in addition to the numbers, dates, or times that appear on the X, Y, or Z axes.

## 4.3.1 Title/Axes command

Enter up to 4 title lines and (depending on scaling options and plot type) one or more labels for the X, Y, and Z axes.

Each of these strings is limited to 200 characters. The appearance of these strings may be changed with the Fonts menu command.

You can embed placeholder codes for certain data parameters (minima, maxima, standard deviation, etc.) in the title lines and axis labels that will be replaced with the corresponding numeric values when the plot is drawn. You can also present the results of equations within the text. For more information on placeholder codes see the <u>Text Placeholders</u> topic.

The **Horizontal** setting for the Y axis label is only applicable to 2D views; it is ignored for 3D views of 3D or 4D data.

Each of these entries may include line breaks so that the text spans multiple lines. To break a string into 2 or more lines, embed a "\n" sequence in the string (for C programmers, that is the actual characters \n, not a linefeed character). For example, this text:

This is the first title\nand this is the 2nd line of the first title

will appear as:

# This is the first title and this is the 2nd line of the first title

The third title line may not be edited while in data processing mode, since this string is continuously updated by DPlot as you edit the data. If you have selected the **DPlot-generated labels** option in data processing mode then you are also prevented from editing the X and Y axis labels.

### Character formatting

Title lines and axis labels may include formatting codes for subscripts, superscripts, etc., as described below. For overbar, dot accents, or Wingdings fonts see *Format Codes* below.

- **X**₂ To use subscripts, select the text (highlight using the cursor) you want to format as subscript, then click this button. **Shortcut:**  $CTRL+\downarrow$
- **x**² To use superscripts, select the text (highlight using the cursor) you want to format as superscript, then click this button. **Shortcut:** CTRL+
- To insert special characters (Greek letters, trademark symbols) found in the Symbol font, click on the insertion point, then click this button. **Shortcut:** CTRL+S

**NOTE:** On versions of Windows older than Vista, the Symbol font with normal weight may only be available in a horizontal orientation. Bold weight can generally be rotated to any angle. This limitation is of course important for the Y axis label on all plot types, and all axis labels in 3D views.

- **B** To make a portion of the text **bold**, select the text (highlight using the cursor) you want to format as bold, then click this button. **Shortcut:** CTRL+B
- To make a portion of the text *italic*, select the text (highlight using the cursor) you want to format as italic, then click this button. **Shortcut:** CTRL+I
- To make a portion of the text <u>underlined</u>, select the text (highlight using the cursor) you want to format as underlined, then click this button. **Shortcut:** CTRL+U
- **t2** To double the size of one or more characters, select the applicable characters (highlight using the cursor), then click this button. **Shortcut:** CTRL+2
- To use a custom color on a portion of the text, select the text (highlight using the cursor) you want to draw using a different color, then click this button. **Shortcut:** CTRL+K

#### Format codes

The formatting buttons simply insert special characters into the text that have specific meaning to DPlot. Use of these buttons is not strictly necessary - you can of course physically type these codes with the same result. Format codes may be nested, so you can have (for example), subscripted Greek characters or subscripts of subscripts. Note that the bold, italic, and underline codes will have no effect if the font already possesses that attribute. In other words, using {\b} on a bold font will **not** make the font "bolder".

Format	Code	Example	Result
Bold	{\btext}	{\bBold} text	Bold text
Italic	{\itext}	{\iltalic} text	Italic text
Subscript	{\dtext}	P{\dso}	P _{so}

			V2 V2
Superscript	{\utext}	X{\u2} + Y{\u2}	X ² + Y ²
Symbol	{\stext}	e{\u(-{\sa}t)}	e ^(-at)
Underline	{_text}	{_underlined} text	underlined text
Enlarged	{ <b>\2</b> text}	{\2{\sS}}x	$\sum x$
Color	{\cBBGGRR <i>text</i> }	{\c0000FFRed} Text	Red Text
Overbar	{\otext}	{\oY}=56.23	<u></u> ¥=56.23
Dot accent	{\. <i>text</i> }	$Flow = \{ \backslash . V \}$	Flow = V
Double dot accent	<b>{\:</b> <i>text</i> <b>}</b>	$y = y{\d0} + {\.y}t - {\.y}t{\u2}/2$	$y = y_0 + \dot{y}t - \ddot{y}t^2/2$
Wingdings	<b>{\w1</b> <i>text</i> <b>}</b>	{\w1N} Danger {\w1N}	& Danger &
Wingdings 2	{\w2text}	{\w2R} Passed	☑ Passed
Wingdings 3	<b>{\w3</b> text <b>}</b>	Arrows {\w3!"#\$} and\nmore arrows {\w3NOPQ}	Arrows ←→1↓ and more arrows ⊊⊃೮೮

## Related macro commands



## 4.3.2 Legend/Labels command

Enter or edit curve legend, labels, or legend title.

You can embed placeholder codes for certain data parameters (minima, maxima, standard deviation, etc.) in the legend entries and legend title that will be replaced with the corresponding numeric values when the plot is drawn. You can also present the results of equations within the text of the legend. For more information on placeholder codes see the <u>Text Placeholders</u> topic.

Each of the legend entries may include line breaks so that the text spans multiple lines. To break a string into 2 or more lines, embed a "\n" sequence in the string (for C programmers, that is the actual characters \n, not a linefeed character).

#### Maximum number of characters

Character strings for the legend are limited to 80 characters. Curve labels are limited to 40 characters, and are placed beside the last data point that is located within the plot window.

#### Legend title

The legend title will be placed above the legend for each of the individual curves. If no legend is specified for any curve, the legend title may still be used to draw a note on the plot.

#### **Curve label**

Curve labels are (usually) short text strings drawn adjacent to the last data point in a curve, or adjacent to the first intersection of a curve with the boundary around the plot. This might be useful when a plot is a bit too crowded to draw a proper legend.

## Example:



#### Hide legend entry for this curve

Check this box to hide an individual legend entry, but not the associated curve. This option might be useful for developing two or more views of a graph in a case where you want to show all curves, but not the complete legend because of its size, for example.

#### Legend line length

Length of the line segment appearing to the left of each legend string. The line is drawn with the same attributes (color, style, and pen width) as the associated curve.

#### Maximum number of rows

Refers to the maximum number of text lines that will be placed in a single column, not including the legend title. Additional columns will be used if the number of lines exceeds this value. DPlot will **not** break a single legend entry containing line breaks (n) into separate columns.

### Number of symbols to use

Specifies how many symbols are drawn in each legend entry (for curves that use symbols). Limited to 0-3. Note that DPlot does nothing to ensure these symbols will not overlap; symbols in the legend are drawn the same size as the corresponding symbols in the plot. The width of the area used to draw symbols is controlled solely by the *Legend line length* entry.

#### Frame legend

Check this box to draw a rectangle around the legend.

#### Opaque

Check this box to hide items that intersect the rectangle bounding the legend.

#### Hide Legend

Check this box to hide (not draw) the legend.

### Location

Enter the X and Y coordinates of the anchor point of the legend, expressed as a fraction of the plot dimensions. (0,0) is at the top left corner of the plot; (1,1) is at the lower right corner. The legend will remain in the same position relative to the plot if you zoom or otherwise change plot extents. Coordinates outside the limits 0-1 are interpreted as legend character sizes. For example (-1,-1) will place the legend anchor point at 1 character width to the left of and 1 character height above the top left corner of the plot.

## **Anchor Point**

You can specify how the box surrounding the legend is aligned with the specified *Location* coordinates.

### Character formatting

The legend title and individual legend character strings may include formatting codes for subscripts, superscripts, etc., as described below. For overbar, dot accents, or Wingdings fonts see *Format Codes* below.

To use subscripts, select the text (highlight using the cursor) you want to format as subscript, then click this button. **Shortcut:**  $CTRL+\downarrow$ 

- To use superscripts, select the text (highlight using the cursor) you want to format as superscript, then click this button. Shortcut: CTRL+
- To insert special characters (Greek letters, trademark symbols) found in the Symbol font, click on the insertion point, then click this button. **Shortcut:** CTRL+S

**NOTE:** On versions of Windows older than Vista, the Symbol font with normal weight may only be available in a horizontal orientation. Bold weight can generally be rotated to any angle. This limitation is of course important for the Y axis label on all plot types, and all axis labels in 3D views.

- **B** To make a portion of the text **bold**, select the text (highlight using the cursor) you want to format as bold, then click this button. **Shortcut:** CTRL+B
- To make a portion of the text *italic*, select the text (highlight using the cursor) you want to format as italic, then click this button. **Shortcut:** CTRL+I

To make a portion of the text <u>underlined</u>, select the text (highlight using the cursor) you want to format as underlined, then click this button. **Shortcut:** CTRL+U

- **t2** To double the size of one or more characters, select the applicable characters (highlight using the cursor), then click this button. **Shortcut:** CTRL+2
- To use a custom color on a portion of the text, select the text (highlight using the cursor) you want to draw using a different color, then click this button. **Shortcut:** CTRL+K

### Format codes

The formatting buttons simply insert special characters into the text that have specific meaning to DPlot. Use of these buttons is not strictly necessary - you can of course physically type these codes with the same result. Format codes may be nested, so you can have (for example), subscripted Greek characters or subscripts of subscripts. Note that the bold, italic, and underline codes will have no effect if the font already possesses that attribute. In other words, using {\b} on a bold font will **not** make the font "bolder".

Format	Code	Example	Result

Bold	{\btext}	{\bBold} text	Bold text
Italic	{\itext}	{\iltalic} text	Italic text
Subscript	{\dtext}	P{\dso}	P _{so}
Superscript	{\utext}	X{\u2} + Y{\u2}	$X^2 + Y^2$
Symbol	{\stext}	e{\u(-{\sa}t)}	e ^(-at)
Underline	{_text}	{_underlined} text	underlined text
Enlarged	{\2text}	{\2{\sS}}x	Σx
Color	{\cBBGGRR <i>text</i> }	{\c0000FFRed} Text	Red Text
Overbar	{\otext}	{\oY}=56.23	<u></u> ¥=56.23
Dot accent	{\. <i>text</i> }	$Flow = \{ \backslash. V \}$	Flow = V
Double dot accent	{\: <i>text</i> }	$y = y{\d0} + {\.y}t - {\.y}t{\u2}/2$	$y = y_0 + \dot{y}t - \ddot{y}t^2/2$
Wingdings	{\w1text}	{\w1N} Danger {\w1N}	≴ Danger &
Wingdings 2	{\w2text}	{\w2R} Passed	⊠ Passed
Wingdings 3	{\w3text}	Arrows {\w3!"#\$} and\nmore arrows {\w3NOPQ}	Arrows ←→1↓ and more arrows ⊊್ರಿರಿಲೆ

## Shortcut:

Double click on the legend.

### Note:

To move the legend, point to the legend and drag to the desired location. To edit an existing legend, you can use this command again, double-click on the legend to bring up this dialog box, or right-click on the legend to change settings.

## 烤 Related macro commands

CurveLabel Legend LegendParams LegendShow

## 4.3.3 Label Points command

Adds a label to a specific data point. For XY plots and 2D views of 3D data you may also add individual labels to points by clicking the *Edit Mode* button on the toolbar.

Edit Mode button:

## XY plots

To label a point near a specific X (or angle for polar plots) value (as opposed to scrolling through the points with the *Point* # horizontal scrollbar), enter the desired value under *Jump to:* and click on the *Go* button. Alternatively, to automatically select the point with the highest or lowest amplitude, click on the *Ymax* or *Ymin* buttons (*Rmax* or *Rmin* for polar plots). Note that these controls are only used to assist you in selecting the point to label; if the coordinates of the labeled point are later changed or if you perform some operation that shifts the minimum or maximum amplitudes, the label is still applied to the point selected in this dialog.

To embed the amplitude of a data point in a label, type **\$Y** in the <u>L</u>abel field. To embed the abscissa of a data point in a label, type **\$X**. When the label is drawn the corresponding numeric values will be
substituted for these control codes. For example, a label of "\$X sec, \$Y psi" at the point (5.5,100) will appear as "5.5 sec, 100 psi". For <u>bubble plots</u> and <u>vector plots</u>, **\$Z** will be replaced with the bubble/vector magnitude. For all other XY plot types, **\$Z** is replaced with "0".

In addition, you can embed placeholder codes for certain data parameters (minima, maxima, standard deviation, etc.) in point labels that will be replaced with the corresponding numeric values when the plot is drawn. You can also present the results of equations within the text. For more information on placeholder codes see the Text Placeholders topic.

The **\$X**, **\$Y**, and **\$Z** codes will use the same number formatting as the corresponding axis. You can restrict the number of decimal places shown in the label by using the <u>ROUND</u> function in an equation. For example, if the Y value of the referenced point is 2.3462567 and the label text is "Y=\$= (ROUND(\$Y,2)), the label will be drawn as "Y=2.35".

**NOTE:** You can quickly add a "\$X, \$Y" label to any point without bringing up this dialog box by right-clicking near the point and selecting "Add \$X, \$Y Label" from the drop-down menu.

### 1D plots (box-and-whisker plots, dot graphs)

Identical to the options for XY plots, except that the "Jump to X" entries are meaningless and therefore disabled.

## **3D surface plots**

To automatically select the point with the highest or lowest amplitude (Z value), click on the Zmax or Zmin buttons. Note that these controls are only used to assist you in selecting the point to label; if the coordinates of the labeled point are later changed or if you perform some operation that shifts the minimum or maximum amplitudes, the label is still applied to the point selected in this dialog.

As with XY plots, the control codes **\$X**, **\$Y**, and **\$Z** will be replaced with the corresponding X, Y, and/or Z components of the labeled point.

Point labels are not drawn on 3D views of surface plots, only on 2D views.

**NOTE:** For all data types, if \$X, \$Y, or \$Z are **not** immediately followed by a space, as in "Y=\$Y kg" vs. "Y=\$Ykg", you should use a backslash character following the \$X, \$Y, \$Z, as in "Y=\$Y\kg". Otherwise the resulting label will not have the desired appearance. The backslash character following \$X, \$Y, or \$Z in the context of point labels has a special meaning: the following characters are irrelevant in interpreting the label text.

A label may include line breaks so that the text spans multiple lines. To break a string into 2 or more lines, embed a "\n" sequence in the string (for C programmers, that is the actual characters \n, not a linefeed character).

If **mouse-activated** is checked, the label will only be drawn when you move the mouse cursor over the data point referenced by the label. The label will not be shown in printed plots or saved images.

To use only the amplitude of the data point as the label, check the "Use Amplitude Only" box.

To move a label, point to the label and drag to the desired location. You may also specify the label location as either an offset in physical units (inches=", centimeters=cm, millimeters=mm, points=pt) or give an absolute location in data space coordinates.

In addition to this menu command, you may edit a label by pointing to the label and double clicking the left mouse button, or by right-clicking on the label and selecting "Edit" from the dropdown menu.

The font used in point labels is the same as that used for the numbers along the axes, and may be changed with the <u>Fonts</u> command on the Text menu.

#### 烤 Related macro commands

DeleteLabel TextPointLabel TextPointLabel3D

### **TextPointLabelLeader**

# 4.3.4 Fonts command

Allows you to change the typeface, color, size, and other attributes of the fonts used to display various labels.

## Shortcut:

Click **Z** on the toolbar.

### Character size

DPlot uses the user-specified character size when: 1) the output device is not a display monitor, or 2) DPlot's window has been maximized. For non-maximized windows, DPlot proportions the size of characters so that a plot will have the same general appearance as a maximized window. If the WYSIWYG option of the <u>Extents/Intervals/Size</u> command is in effect, then fonts are scaled down, if necessary, so that a plot on the display will have the same appearance as a printed copy.

#### Typeface

By default, DPlot uses a TrueType Arial font for all text.

Be sure that you have enabled TrueType fonts using the **Fonts** option of **ControlPanel** before running DPlot. TrueType fonts will generally provide the best performance across multiple output devices, and are the only standard fonts that can be rotated reliably.

If printing to a pen plotter, DPlot selects the plotter's stick font, regardless of which typeface you have chosen.

#### Saving font options

On exit from DPlot, font information is stored in the initialization file **DPLOT.INI**, found in the Application Data\DPlot folder. If **DPLOT.INI** is erased or otherwise damaged, DPlot will use the default TrueType Arial font until you select new fonts. Font point sizes are saved to a DPlot data file.

## 烤 Related macro commands

FontPoints TextFont

# 4.3.5 Add/Edit Note command

Enter or edit a notation, placed at an arbitrary location on the plot. Each notation is limited to 4095 characters. There is no limit to the number of notations per plot, though of course there are practical limits due to limited space on the plot.

Unlike most DPlot dialogs, pressing ENTER while the <u>Note</u> control has the input focus results in a new line in the notation (normally, pressing ENTER is equivalent to clicking on OK). Press CTRL+TAB to embed a TAB character in the note. Press TAB to shift the input focus to another control.

To paste the contents of the Clipboard to the notation, move the cursor to the desired insertion point and press CTRL+V, or right-click and select *Paste*.

**Location and Units:** Enter the X and Y coordinates of the anchor point of the note, expressed as a fraction of the plot dimensions (Units=Plot space) or in the same units as the X and Y axes (Units=Data space).

#### Units=Plot space

(0,0) is at the top left corner of the plot; (1,1) is at the lower right corner. The notation will remain in the same position relative to the plot if you zoom or otherwise change plot extents. Coordinates outside the limits 0-1 are interpreted as note character sizes. For example (-1,-1) will place the note anchor point at 1 character width to the left of and 1 character height above the top left corner of the plot.

#### Units=Data space

The note position is tied to your data. If you zoom or change the plot extents, the note position relative to the plot may change. If the note location is outside the extents of the plot, it will not be drawn. If you drag a note outside the plot extents unintentionally, or make a mistake entering the position, you can correct this by opening the Add/Edit Note dialog again and entering location values that are within the plot extents.

You may also move a notation by pointing to the bounding rectangle and dragging to the desired location. If Units=Plot space and you drag a note such that the anchor point is outside the plot extents, it will not be drawn.

#### **Anchor Point**

You can specify how the note is aligned with the specified *Location* coordinates.

#### **Text Placeholders**

You can embed placeholder codes for certain data parameters (minima, maxima, standard deviation, etc.) in each notation that will be replaced with the corresponding numeric values when the plot is drawn. You can also present the results of equations within the text. For more information on placeholder codes see the Text Placeholders topic.

#### Font

The font used in notations may be changed with the <u>Fonts</u> menu command or by clicking the *Font>>* button.

#### **Character formatting**

Notations may include formatting codes for subscripts, superscripts, etc., as described below. **Please note:** Formatting codes should **not** extend across multiple lines of text, whether they are entered via the buttons or manually. For overbar, dot accents, or Wingdings fonts see *Format Codes* below.

**X**₂ To use subscripts, select the text (highlight using the cursor) you want to format as subscript, then click this button. **Shortcut:**  $CTRL+\downarrow$ 

- X² To use superscripts, select the text (highlight using the cursor) you want to format as superscript, then click this button. Shortcut: CTRL+
- α To insert special characters (Greek letters, trademark symbols) found in the Symbol font, click on the insertion point, then click this button. **Shortcut:** CTRL+S

**NOTE:** On versions of Windows older than Vista, the Symbol font with normal weight may only be available in a horizontal orientation. Bold weight can generally be rotated to any angle. This limitation is of course important for the Y axis label on all plot types, and all axis labels in 3D views.

- **B** To make a portion of the text **bold**, select the text (highlight using the cursor) you want to format as bold, then click this button. **Shortcut:** CTRL+B
- **I** To make a portion of the text *italic*, select the text (highlight using the cursor) you want to format as italic, then click this button. **Shortcut:** CTRL+I
- To make a portion of the text <u>underlined</u>, select the text (highlight using the cursor) you want to format as underlined, then click this button. **Shortcut**: CTRL+U
- **t2** To double the size of one or more characters, select the applicable characters (highlight using the cursor), then click this button. **Shortcut:** CTRL+2
- To use a custom color on a portion of the text, select the text (highlight using the cursor) you want to draw using a different color, then click this

#### button. Shortcut: CTRL+K

#### Format codes

The formatting buttons simply insert special characters into the text that have specific meaning to DPlot. Use of these buttons is not strictly necessary - you can of course physically type these codes with the same result. Format codes may be nested, so you can have (for example), subscripted Greek characters or subscripts of subscripts. Note that the bold, italic, and underline codes will have no effect if the font already possesses that attribute. In other words, using {\b} on a bold font will **not** make the font "bolder".

Format	Code	Example	Result
Bold	{\btext}	{\bBold} text	Bold text
Italic	{\itext}	{\iltalic} text	Italic text
Subscript	{\dtext}	P{\dso}	P _{so}
Superscript	{\utext}	X{\u2} + Y{\u2}	$X^2 + Y^2$
Symbol	{\stext}	e{\u(-{\sa}t)}	e ^(-at)
Underline	{_text}	{_underlined} text	underlined text
Enlarged	{\2 <i>text</i> }	{\2{\sS}}x	Σx
Color	{\cBBGGRR <i>text</i> }	{\c0000FFRed} Text	Red Text
Overbar	{\otext}	{\oY}=56.23	<u></u> ¥=56.23
Dot accent	<b>{\.</b> <i>text</i> <b>}</b>	$Flow = \{ \backslash. V \}$	Flow = V
Double dot accent	{\: <i>text</i> }	$y = y{\d0} + {\.y}t - {\.y}t{\u2}/2$	$y = y_0 + \dot{y}t - \ddot{y}t^2/2$
Wingdings	<b>{\w1</b> <i>text</i> }	{\w1N} Danger {\w1N}	≵ Danger <b>≵</b>
Wingdings 2	{\w2text}	{\w2R} Passed	☑ Passed
Wingdings 3	<b>{\w3</b> text <b>}</b>	Arrows {\w3!"#\$} and\nmore arrows {\w3NOPQ}	Arrows ←→1↓ and more arrows ⊊೨೦೮

#### Frame note

Check this box to draw a rectangle around the notation.

#### Opaque

Check this box to hide items that intersect the rectangle bounding the notation.

#### **Button activated**

Check this box to display a pushbutton in place of the note. Clicking on the pushbutton displays a popup window containing the note. If this option is selected the note will <u>not</u> be printed or copied to the Clipboard by the <u>Copy Picture</u> command. The text appearing on the note button is controlled by the **Button Text** edit box. If this box is left blank the button will contain the word "Note".

## Tab Stops at

Enter the number of characters for each tab stop, or click on the arrows until the desired tab spacing is selected. Tab characters may be embedded in the note by typing CTRL+TAB.

#### Draw arrow to

Check this box to draw a leader from the notation to a point on the plot, specified in data space in the X and Y boxes below. Control the line width and arrowhead size with the adjacent **Arrow dimensions** button. If either the X or Y text boxes have the keyboard focus (denoted by the flashing text cursor in either box) then you can click on the plot to specify those coordinates. If **Snap cursor to nearest data point** is checked **and** you click near a curve, the X and Y boxes will be filled with the coordinates of the closest data point.

#### Shortcut:

Double click on an existing note.
 or
 Click on the toolbar.
 Related macro commands
 DeleteNote
 NoteButton Text
 NoteOptions
 NotePosition
 NoteTabstops

TextNote TextNote3D TextNoteEx TextNoteLeader

# 4.3.6 Text Placeholders

You can use placeholder codes in a plot's title lines, axis labels, legend, and notations that will be replaced with the corresponding numeric or text values when the plot is drawn. Codes that are replaced with numbers may also be used in equations with the  $\underline{Y=f(X)}$  and similar commands, as well as in equations within <u>macro/DDE commands</u>. Many codes may be specific to a particular data set or curve, or to the entire plot. All codes are preceded by a dollar sign (\$). Placeholder codes and the result of their use is shown below.

## Code **Replaced with** \$AREA Area under a curve for XY plots. Calculated using the trapezoidal rule: $A = \frac{1}{2} \sum_{j=2}^{n} (y_j + y_{j-1}) \times (x_j - x_{j-1})$ For 3D and 4D surface plots, \$AREA reports the total surface area. \$FILETITLE Text appearing in the window caption of the document, normally the filename of the document if it has been saved. \$LENGTH Dimensionless length of a curve (XY plots only). $L = \sum_{i=2}^{N} \sqrt{(x_j - x_{j-1})^2 + (y_j - y_{j-1})^2}$ \$MEAN Average amplitude. For 2D data, the average Y value. For 3D data, the average Z value. **\$NEDGE** Number of exterior edges (3D plots only) \$NC 1-based curve index. This shortcut is generally only useful when used in the legend or with the Operate on X or Operate on Y commands on the Edit menu. For example, with Operate on Y, Y=Y+\$NC will add 1 to all Y values in the first curve, 2 to all Y values in the second curve, etc.

- \$NPTS Number of data points
- \$NTRI Number of triangles (randomly-spaced 3D points only)

\$RMS Root-Mean-Square (RMS) of the data amplitudes (Y values for 1D and 2D plots, Z values for 3D data). The RMS is defined by:

$$rms = \sqrt{\frac{\sum {x_j}^2}{N}}$$

where

x = amplitude

N = number of data points

\$SUMX Sum of the X values for a curve. (XY plots only)

- \$SUMY Sum of the Y values for a curve. (XY plots only). Barring overflow errors, should be equal to \$MEAN * \$NPTS for a given curve.
- \$STDEV Standard deviation in the data amplitudes. For 2D data, standard deviation in the Y values. For 3D data, the standard deviation in the Z values.

Standard deviation is defined by

$$stdev = \sqrt{\frac{\sum (x - \overline{x})^2}{N - 1}}$$

where

x = amplitude

x = mean

N = number of data points

For MS Excel users, this returns the same value as Excel's STDEVA function.

- \$X(i,j) X value of the j'th point of the i'th curve. Only meaningful for XY plots. For example, \$X(3,12) will be replaced with the X value of the 12th point from the 3rd curve.
- \$XMIN Minimum X value. Ignored for box-and-whisker plots and dot graphs.
- \$XMAX Maximum X value. Ignored for box-and-whisker plots and dot graphs
- \$XRANGE Maximum X value minus the minimum X value.
- \$Y(i,j) Y value of the j'th point of the i'th curve. Only meaningful for XY plots. For example, \$Y(3,12) will be replaced with the Y value of the 12th point from the 3rd curve.
- \$YMIN Minimum Y value.
- \$YMAX Maximum Y value.
- \$XYMIN XY plots only. The X coordinate where the minimum Y value occurs.
- \$XYMAX XY plots only. The X coordinate where the maximum Y value occurs.
- \$YRANGE Maximum Y value minus the minimum Y value.
- \$YFIRST Y value of the first point in a curve (XY plots only).
- \$YLAST Y value of the last point in a curve (XY plots only).

## 3D/4D plots only:

\$ZMIN Minimum Z value.

\$ZMAX	Maximum Z value.
\$ZRANGE	Maximum Z value minus the minimum Z value.
\$VZMIN	Minimum visible Z value after zooming or setting extents with Extents/Intervals/Size
\$VZMAX	Maximum visible Z value after zooming or setting extents.
\$VZAVG	Average Z value after zooming or setting extents.
\$VXMIN	Minimum X value after zooming or setting extents.
\$VXMAX	Maximum X value after zooming or setting extents.
\$VYMIN	Minimum Y value after zooming or setting extents.
\$VYMAX	Maximum Y value after zooming or setting extents.
\$= ( <equation>)</equation>	Result of evaluating <equation>. For example, "\$=(sqrt(5))" will be replaced with "2.23607". Note that the equation <b>must</b> be enclosed by parentheses. Other text shortcuts that are replaced by a number (i.e. any shortcut other than \$FILETITLE) may be used in an equation, as in "\$=(\$STDEV(1)/\$MEAN(1))".</equation>
	For trigonometric functions, the argument is always assumed to be in <b>degrees</b> . This differs a bit from $Y=f(X)$ and other commands that allow a choice between degrees and radians.
	For more information on equation syntax, operators, and

supported functions see the description of the Y=f(X) command.

Placeholder text IS case-sensitive, e.g. \$Xmax will be ignored.

Many of the placeholders may include a numeric argument specifying which data set the placeholder refers to. That numeric argument must be enclosed with parentheses. A 0 argument means that the value for the entire plot (rather than a specific data set) should be used. If the data set argument is omitted, then a context-sensitive decision is made by DPlot about whether the placeholder refers to a specific data set or to the entire plot. For example,

\$YMAX when used in a title line or notation will be replaced with the maximum Y value for **all** data sets. If used in a legend entry, then it will be replaced with the maximum Y value of the data set (curve) associated with that legend entry. To use the maximum Y value for **all** data sets in a legend entry, then, you must use \$YMAX(0) rather than \$YMAX.

\$YMAX(5) will always be replaced with the maximum Y value from the 5th data set, regardless of where this placeholder is used.

Area (\$AREA), mean (\$MEAN) and standard deviation (\$STDEV) values for aggregate multiple data sets are never calculated. If you use a 0 argument for these values then the placeholder will be replaced with the area, mean or standard deviation for the first data set.

## Range

For the \$AREA (2D only), \$MEAN (2D only), \$YMIN, \$YMAX, \$ZMIN, and \$ZMAX placeholders you may include a range of X (or X and Y for \$ZMIN, \$ZMAX) within which to find those values.

AREA(<<</br>

 \$AREA(<<</td>
 number>,<<</td>
 xlo>,<<</td>
 xhi>)
 returns the area under curve number <</td>
 curve number> that lies

 between <<</td>
 xlo> <</td>
 X <</td>
 <</td>
 xhi>)

MEAN(< number>,<xlo>,<xlo>,<xhi>) returns the average amplitude for curve number <curve number> that lies between <xlo>  $\leq X \leq$  <xhi>. This applies to 2D data only.

 $MIN(<curve number>,<xlo>,<xhi>) returns the minimum Y value for curve number <curve number> that lies between <xlo> <math>\leq X \leq <xhi>$ .

**Example:** YMIN(2,1.6,3.5) will be replaced with the minimum Y value between  $1.6 \le X \le 3.5$  in the 2nd

curve. If no points lie between those extents, the replacement text is "!none found".

MAX(< number>, < xlo>, < xhi>) returns the maximum Y value for curve number < curve number> that lies between < xlo>  $\leq$  X  $\leq$  < xhi>.

\$ZMIN(<data set>,<xlo>,<ylo>,<xhi>,<yhi>) returns the minimum Z value that lies within the bounds <xlo>,<ylo> to <xhi>,<yhi>.

\$ZMAX(<data set>,<xlo>,<ylo>,<xhi>,<yhi>) returns the maximum Z value that lies within the bounds <xlo>,<ylo> to <xhi>,<yhi>.

In the context of specifying a range, \$YMIN and \$YMAX are valid only for XY plots. \$ZMIN and \$ZMAX are valid only for 3D surface plots and scatter plots. In all cases, the <curve number>/<data set> **must** be specified. For 3D surface plots this value should always be 1. For XY plots, if <curve number> is set to 0 then the check is against **all** curves. Similarly, for 3D scatter plots if <data set> is set to 0 then the check is for all data sets.

The range values must be specified using numbers only, not equations and/or other text placeholders.

# 4.3.7 X,Y,Z Labels

This command allows you to specify up to 100 text labels that can replace or be used in combination with the numbers, dates, and/or times that appear on the X or Y axes or the Z axis of a 3D or 4D plot. No effort is made to prevent these labels from overlapping those numbers, dates, or times. The labels will be drawn with the same orientation as the numbers along the axis. Orientation is controlled with the <u>Number Formats</u> command on the Options menu. To use these labels **only**, set the number format for the corresponding axis to "None" using the Number Formats command. Alternatively, right-click on any of those numbers and select "None".

Similar functionality can be achieved with <u>notes</u>, but exact placement of notes is a bit cumbersome: If note placement is made a function of plot space (the default), if you zoom or otherwise change the extents of the plot the note positions will change. And if note placement is made a function of data space, the note will not be drawn if the anchor point is outside the limits of the plot.

X/Y/Z value: The location of the label along the axis chosen with Location (see below).

Label: The text you want to draw.

- **Location:** The axis associated with the label. For 3D views of 3D or 4D data, the labels will always be drawn on the same side of the plot that the numbers along the coordinate axes would normally be drawn.
- Offset: The distance from the alignment point of the normal axis labels to the alignment point of the label you're creating. In most cases 0 is the appropriate value. But if you want, for example, to call attention to a particular label you might want to give a small positive offset. Offsets are positive away from the plot: A positive offset for "Y left" shifts the label to the left, for "Y right" to the right.

Example:



Joe's Fruit Stand Sales

For a step-by-step explanation of how this graph was created, see the topic <u>How do I replace the X (or</u> <u>Y) axis labels with custom text?</u>

## Character formatting

Labels may include formatting codes for subscripts, superscripts, etc., as described below. For overbar, dot accents, or Wingdings fonts see *Format Codes* below.

- To use subscripts, select the text (highlight using the cursor) you want to format as subscript, then click this button. **Shortcut:**  $CTRL+\downarrow$
- **x**² To use superscripts, select the text (highlight using the cursor) you want to format as superscript, then click this button. **Shortcut:** CTRL+
- To insert special characters (Greek letters, trademark symbols) found in the Symbol font, click on the insertion point, then click this button. **Shortcut:** CTRL+S

**NOTE:** On versions of Windows older than Vista, the Symbol font with normal weight may only be available in a horizontal orientation. Bold weight can generally be rotated to any angle. This limitation is of course important for the Y axis label on all plot types, and all axis labels in 3D views.

- **B** To make a portion of the text **bold**, select the text (highlight using the cursor) you want to format as bold, then click this button. **Shortcut:** CTRL+B
- To make a portion of the text *italic*, select the text (highlight using the cursor) you want to format as italic, then click this button. **Shortcut:** CTRL+I
- To make a portion of the text <u>underlined</u>, select the text (highlight using the cursor) you want to format as underlined, then click this button. **Shortcut:** CTRL+U
- **t2** To double the size of one or more characters, select the applicable characters (highlight using the cursor), then click this button. **Shortcut:** CTRL+2

To use a custom color on a portion of the text, select the text (highlight using the cursor) you want to draw using a different color, then click this button. **Shortcut:** CTRL+K

#### Format codes

龖

The formatting buttons simply insert special characters into the text that have specific meaning to DPlot. Use of these buttons is not strictly necessary - you can of course physically type these codes with the same result. Format codes may be nested, so you can have (for example), subscripted Greek characters or subscripts of subscripts. Note that the bold, italic, and underline codes will have no effect if the font already possesses that attribute. In other words, using {\b} on a bold font will **not** make the font "bolder".

Format	Code	Example	Result
Bold	{\btext}	{\bBold} text	Bold text
Italic	{\itext}	{\iltalic} text	Italic text
Subscript	{\dtext}	P{\dso}	P _{so}
Superscript	{\utext}	X{\u2} + Y{\u2}	$X^2 + Y^2$
Symbol	{\stext}	e{\u(-{\sa}t)}	e ^(-at)
Underline	{_text}	{_underlined} text	underlined text
Enlarged	{\2 <i>text</i> }	{\2{\sS}}x	Σx
Color	{\cBBGGRR <i>text</i> }	{\c0000FFRed} Text	Red Text
Overbar	{\otext}	{\oY}=56.23	<del>▼</del> =56.23
Dot accent	<b>{\.</b> <i>text</i> <b>}</b>	$Flow = \{ \backslash. V \}$	Flow = V
Double dot accent	{\: <i>text</i> }	$y = y{\d0} + {\.y}t - {\.y}t{\u2}/2$	$y = y_0 + \dot{y}t - \ddot{y}t^2/2$
Wingdings	{\w1 <i>text</i> }	{\w1N} Danger {\w1N}	& Danger &
Wingdings 2	<b>{\w2</b> text <b>}</b>	{\w2R} Passed	☑ Passed
Wingdings 3	<b>{\w3</b> text <b>}</b>	Arrows {\w3!"#\$} and\nmore arrows {\w3NOPQ}	Arrows ←→1↓ and more arrows ⊊೨೮೮

#### 烤 Related macro commands

**XYLabel** 

# 4.4 Options Menu

#### Linear/Log Scaling

Change how the data is scaled to device.

## **Grid Lines or Box**

Controls the appearance of the X and Y axes.

## **Scatter Plot**

Set all line styles to "None" and symbol styles (for curves not currently using symbols) to filled circles. For more control over line and symbol styles, use:

## Symbol/Line Styles

Change the line styles, line widths, symbol style and symbol sizes used in the graph.

## Extents/Intervals /Size

Set the extents of the plot, number of tick marks and/or tick mark interval, and the physical size of

the plot.

## **Tick Marks**

Controls the placement (inside, outside, or both) and length of minor and major tick marks.

#### Square Log Cycles

Toggles state of the appearance of log cycles.

## Log Scale Tick Labels

Determines frequency of tick mark labels on logarithmic scales.

#### Multiple Axes

Presents several choices for scaling the X and Y axes.

#### **Number Formats**

Allows you to change the appearance of numbers.

#### <u>Colors</u>

Change color palettes used for drawing curves.

#### Fill Between Curves

Draws a hatched pattern or solid fill between two curves.

#### **Bubble Plot**

Set source curve for bubble magnitudes, and set bubble color, size, and symbol used.

#### Vector Plot

Set parameters for vector plot (4 dimensional plot: x,y,angle,magnitude).

#### **OHLC Chart**

Set parameters for an Open-High-Low-Close chart (primarily used for stock market data).

#### Error Bars

Set parameters for error bars for one or more curves.

#### **Reference Lines**

Draw horizontal or vertical lines at specified X or Y values

## Amplitude Limits

Specify amplitude range outside which points are ignored, potentially creating gaps in your curves.

#### Background Image

Allows you to specify a bitmap image to draw in a graph's background covering the entire plot extents.

#### Insert Image

Specify up to 4 clipart images to be placed at various locations around or within the plot.

#### **Data Processing**

Automatically generate integrations and set plotting options.

#### **General**

Odds and ends that don't fit anywhere else. Always maximize DPlot, prompt to save changed plots, and allow undo.

## **Contour Plots Only**

#### **Contour Options**

Control the appearance of contour plots

#### Generate Mesh

Generate a rectangular grid of data values from the current 3D surface plot

#### **Define Boundary**

Allows you to define a concave exterior boundary and/or holes in a surface.

## Set Limits to Min/Max in Box

Drag the cursor across the plot to define a box. The minimum and maximum Z values within that box

will then be used as the lower and upper contour intervals. Disabled in 3D views.

**3D Scatter Plots Only** 

#### **Scatter Plot Options**

Control the appearance of 3D scatter plots

#### **1D Statistics Plots Only**

#### Box-and-Whisker Plot/Dot Graph

Both the Box-and-Whisker Plot and Dot Graph serve as 2-dimensional displays to visually compare the mean or median values and variability of several groups. In essence, the display gives the user a sense for the distribution of values for each group.

## Format

Controls the parameters and attributes of statistical plots.

#### 💋 Related macro commands

Stat_AddData Stat AddGroup Stat_Box Stat_BW Options Stat DotOptions Stat DrawLegend Stat_GrandMean Stat GroupInfo Stat_LimitsAsCL Stat_MeanFormat Stat Outliers Stat_PlotType Stat RefLine Stat Rotated Stat_Scaling Stat Whiskers

# 4.4.1 Linear/Log Scaling

Available scaling options for XY plots are:

# □ Linear X, Linear Y

This is the default scaling for all new XY plots.





□ Linear X , Logarithmic Y Y values must be  $\geq$  0. Data points with Y=0 are ignored.



□ Logarithmic X, Linear Y X values must be  $\geq 0$ . Data points with X=0 are ignored.









Amplitudes for probability scales must be in the range 0 < y < 100, or optionally 0 < y < 1. Values less than 0.00001 (1.E-5) or greater than 99.99999 (1.E-7 or 0.99999999 for a 0-1 range) are clipped. A Cumulative Distribution Function for a normally distributed variable will plot as a straight line using "Linear X - Probability Y" scaling:



# **Cumulative Distribution Function**

Tripartite Grid

Tripartite grids are generally used to plot shock spectra for a given velocity record or fragility curves for shock-sensitive equipment. By default, a tripartite grid is set up for a shock spectra plot with X values = Frequency in Hertz, Y values = velocity in inches/sec, 45 degree lines = displacement in inches, and 135 degree lines = acceleration in g's. Units, the density of the acceleration and displacement lines, and the frequency of the acc./disp. labels may be changed with the Tripartite Options menu command.



The tripartite grid selection will override the grid type selection made with the <u>Grid Lines or Box</u> command.

## Grain Size Distribution

Amplitudes (percent finer values) for "Grain Size Distribution" scaling must be in the range 0 < y < 100, and the abscissa (grain sizes) must consist of positive values. Unlike all other scaling options, this option automatically fills in values for the X and Y axis labels, as well as a second Y axis label located on the right side of the plot. These labels may be changed, however, with the Title/Axes menu option.



By default, the grain size plot is set up for grain sizes in millimeters. A U.S. Standard Sieve scale will be drawn at the top edge of the plot, and a scale showing the range in size of common engineering materials is drawn below the X axis. These features require that the units of the grain sizes be known. The units may be changed with the <u>Grain Size Options</u> menu command.

Grain size distribution plots will generally have a better appearance if you force the abscissa (X) to range from 0.001 mm to 1000. mm (or 0.00001 to 100. inches) using the <u>Manual Scaling</u> menu option.

The grid type selection made with Grid Lines or Box is ignored.

## Polar Coordinates



By default, angles are assumed to be in radians. Assumed angular units and the orientation of the plot may be changed with the Polar Plot Options menu command. Angles are displayed in degrees regardless of which unit type is in effect, unless the "Pi Multiples" or "Pi Fractions" <u>number format</u> is used for the X axis.

The grid type selection made with Grid Lines or Box is ignored.

#### Bar Chart



Floor-Load Data

With multiple data sets, the abscissa must have a constant spacing for bar charts. If only one data set is used then the abscissa may have unequally-spaced values. X values are used as the lower limit of each bar, and the width of the bar is set to some fraction (default=1.0) of the spacing between points. The appearance of bar charts (filled or hollow, stacked or side-by-side, labeling bar amplitudes) may be controlled using the Bar Chart Options command.

If you want the bars to begin at the associated X value, you can modify X to accomplish this. Use Edit>Operate on X with X=X-(X(1,2)-X(1,1))/2 to shift the bars left by 1/2 the distance in X between the first 2 points in the first data set.

## Triangle Plot

Triangle plots are useful for plotting various mixtures with 3 components (for example, soil is made up of sand, silt, and clay). The three components of each point (x, y, and z) always add up to 100. Since the components are therefore not independent, the points are completely determined by the x and y values. For more information on creating triangle plots see the topic for the <u>example plot</u> EX14.GRF.

## Soil Composition



As with polar coordinates, several menu options are disabled when viewing triangle plots. In particular, you cannot use the Zoom command or the <u>Extents/Intervals /Size</u> command to modify the plot extents. Also see the <u>Always force symbols on/lines off for triangle plots</u> option of the Options>General command.

#### ■ N1.85 Hydraulic curve

This graph type is typically used in hydraulic studies, particularly for fire suppression systems. The horizontal coordinates of the graph are scaled to the 1.85 power because, in the Hazen-Williams formula, pressure is proportional to the flow to the 1.85 power.



# Hazen-Williams Formula for Pipe Friction Loss 100 foot plastic pipe

## Mercator Projection

This scaling type is typically used for maps. Input values are limited to  $-360 \le x \le 360$  and  $-85 \le y \le 85$ , where x is longitude in degrees and y is latitude in degrees. (DPlot will force the <u>Amplitude</u> <u>Limits</u> setting on for Mercator Projection and set the limits to +/- 85, so your input **may** contain latitudes outside those limits, that data will just be ignored.)

Although any <u>number format</u> may be used with this scaling, it is designed for *Degrees, Minutes* or *Degrees, Minutes, Seconds* and has no practical value for other formats.



The vertical axis with this scaling is increasingly stretched out at higher latitudes. This scaling was originally made popular by its ability to represent lines of constant true bearing as straight line segments.

DPlot will automatically amend either the width or height of the plot such that the correct scaling is used. It will make this change, if necessary, regardless of whether the *Specify size* option of the <u>Extents/Intervals/Size</u> command is used. For predictable results with this scale type, use Extents/ Intervals/Size, check the *Specify size* option and enter the desired width and height of the plot, then click the *Set X:Y=1:1* button.

Although DPlot will display longitudes as expected (-200° is displayed as 160°, for example), on input the data will be drawn with decreasing x values (longitudes) to increasing x values from left to right. Therefore if your map crosses longitude 180° and your input longitudes are positive in the Eastern hemisphere, negative in the Western hemisphere (as is standard practice), you will need to modify the longitudes for the map to be drawn correctly. This is easily done with the <u>Operate on X</u> command on the Edit menu with, for example, X=IF(X>0,X-360,X).

## Shortcut:

Right-click within the graph.

# 💋 Related macro commands

BarChartStyle BarWidth FillStyle PolarPlotStyle GrainSizeFlags GrainSizeFlagsEx GrainSizeOther ProbabilityFlags Scale TriangleFlags TripartiteFlags

#### See also: Text Menu

## 4.4.1.1 Triangle Plot Options

Controls the appearance of a triangle plot.

# Shortcut:

Right-click within an existing triangle plot and select Triangle Plot Options.

# **Clockwise Orientation**

Reverses the normal counterclockwise orientation of the numbers on the axes. Using this option the Y components will be mapped to the scale on the left, Z to the right.



## X+Y+Z=1

By default DPlot forces X+Y+Z=100. Check this box if the range on each of the 3 components is 0-1.0.

## **USDA Soil Classification**

If this option is checked, a triangle plot will be divided into the 12 soil texture classes defined by the US Department of Agriculture. The scaling assumes that the data's X components (bottom scale) are sand percentage (0-100, or 0-1 if X+Y+Z=1 is checked) and Y components (right side of triangle) are clay percentage. The left side of the triangle is then silt percentage, forced to 100-(X+Y), or to 1.0-(X+Y) if X+Y+Z=1 is checked. If your data is instead X=sand and Y=silt percentages, to properly use this option you would first need to perform the transform Y=<range>-(X+Y) using the *Operate on* Y command on the Edit menu, where <range>=1.0 if X+Y+Z=1 is checked, and 100 otherwise.



## **IUGS Mafic Rock Classification**

Draws boundaries and labels for IUGS (International Union of Geological Sciences) mafic rock. X axis is Olivine percentage, Y axis is Plagioclase percentage, and the inferred Z axis is Clinopyroxene percentage.



## **IUGS Ultramafic Rock Classification**

Draws boundaries and labels for IUGS (International Union of Geological Sciences) ultramafic rock classifications. X axis is Clinopyroxene percentage, Y axis is Olivine percentage, and the inferred Z axis is Othopyroxene percentage.



### **Ab-An-Or Mineral Classification**

Draws boundaries and labels for Albite-Anorthite-Orthoclase classification. X values are assumed to be Orthoclase percentage, Y values are assumed to be Anorthite percentage, and the inferred Z values are Albite percentage.



**Note:** Most moderate-sized non-bold fonts will have a pronounced aliasing problem at orientations other than horizontal or vertical on your display. They're ugly, in other words. For triangle plots this pertains to the numbers along the X and Z axes. The equivalent **bold** font will generally have a better appearance in this case. This applies mainly to the display and saved bitmaps; printed fonts will generally look good regardless of the orientation or font weight.

## Selated macro commands

TriangleFlags

# 4.4.1.2 Polar Plot Options

This command allows you to change the appearance of polar plots.

# Shortcut:

Right-click within an existing polar chart and select Polar Plot Options.

#### Angular units

Select the units for the independent variable. NOTE: Changing units does NOT alter the input data

in any way.

## 0 degrees at top of plot

By default, 0 degrees is drawn at the 3 o'clock position.

## **Clockwise angles**

If checked, angular values increase in a clockwise direction.

## Force circular plots

If unchecked, then the plot will be drawn as large as possible, subject to the "Specify size" setting under <u>Options>Extents/Intervals /Size</u>.

If checked, the width and height of the plot are forced to be equal, which results in a circular plot. If the size has not been specified then it will be set to roughly 80% of the minimum of the document window width and height. You can override that size selection with the "Specify size" setting under <u>Options>Extents/Intervals /Size</u>.

## Vertical radial axis labels

If checked, labels for the dependent variable will have a vertical orientation.

## Center radial labels on the radius

If checked, labels for the dependent variable will be centered on the corresponding radius. If unchecked, labels will be left-justified (0 degrees at 3 o'clock) or bottom-justified (0 degrees at top of plot).

## Label 0 degrees

If unchecked, the 0 degrees label is not drawn, and the maximum radius label is drawn at or near that location. If checked, the 0 degrees label is drawn and the maximum radius label is not.

## Axial grid lines start at

By default, axial grid lines emanate from the center of the plot. With a large number of grid lines (set with the Tick Marks settings under <u>Options>Extents/Intervals /Size</u>) and/or low resolution output, this may result in undesirable visual artifacts or a cluttered appearance.

#### Angles from 0 to +/- 180

Displays angular values as 0-180 and 360-angle for angles between 180 and 360 degrees.

# Transform negative radii to angle+PI, abs(radius)

For negative radii, sets angle values to angle+180 degrees and radius=abs(radius). Note that unlike all other polar settings this option actually operates on the data and is therefore not reversible. The value added to the angle (X) coordinates is either 180 for angles in degrees or PI for angles in radians. Since this is a one-time operation it is important that you establish the correct angular units before using this option.

## Related macro commands

**PolarPlotStyle** 

## 4.4.1.3 Bar Chart Options

# Shortcut:

Right-click within an existing bar chart and select Bar Chart Options.

## Stacked bars

Check this box to draw bars stacked from bottom to top.



Leave this box unchecked to draw bars side-by-side.



## Amplitude labels

Check this box to draw the amplitude of each bar near the top of the bar.



#### **Percentage labels**

Check this box to draw the amplitudes of each bar as a percentage of the total count for the given data set. Check the "For ALL data sets" box if you instead want a percentage of the total count for all data sets.



#### Labels outside bars

Check this box to draw labels (selected above) above the top of each bar, rather than on the bar.



## Bars span intervals in X (1 data set only)

Check this box if your data values are unevenly-spaced on the X axis and you want the bars to extend from the given X to the X value for the next point. In this case the last data point in a set is ignored, other than to define the boundary for the bar associated with the next-to-last point. This option overrides the *Bar width* slider setting, and is ignored if the bar chart has more than one data set.



## Unique colors for each bar (1 data set only)

Check this box to use a different color for each bar. The color assigned to curve #1 (with the <u>Colors</u> command) is used for the first bar, and in general the color assigned to curve N is used for the Nth bar. This option is ignored if the bar chart has more than one data set.

## Base of bars at minimum Y extent

If checked, bars are drawn with the base at the lower extent of the graph.



If unchecked, bars are drawn with the base at Y=0:



This option is ignored if "Stacked bars" is checked.

## No black border on filled bars

If checked, only the filled area of the bars is drawn; there will be no border around the bars. This option is ignored if "Outline extremes only" is checked.



## Outline extremes only (1 data set only)

If checked, vertical edges between adjacent bars are not drawn. If this option is selected then "No black border..." is ignored. This option is only valid for a single data set, and is ignored if "Unique colors for each bar" is checked or if either the bar width or group width (see below) is less than 100.



## Shaded bars (solid fill pattern only)

For solid bars, shades the interior as if the bars are cylindrical.



## Horizontal bars

Check this box to draw horizontal bars.



#### Bar width and Group Width

These values control the width of the bars and the space between bars with the same ordinate. Use 100% to fill the space between input X values. Examples:

Bar width=75%, Group width=100%:



Bar width=100%, Group width=75%:



Bar width=75%, Group width=75%:



#### Fill patterns

"Hollow" bars will be outlined using the color associated with the data. All other fill styles will be outlined in black (subject to the "No black border" and "Outline extremes" settings), with the interior filled with the selected pattern in that same color.

## 烤 Related macro commands

BarChartStyle BarWidth FillStyle

## 4.4.1.4 Grain Size Options

This command allows you to change the appearance of grain size distribution plots.

# Shortcut:

Right-click within an existing grain size distribution plot and select Grain Size Options.

#### Specify sieve size labels

If unchecked, DPlot will start drawing labels at the largest size that is within the extents of the plot, and subsequently draw only those labels which will not overlap previously-drawn labels. Note that the result of this behavior will be dependent on the font size used for the numbers along the axes (also used for sieve labels) and the physical size of the plot. If checked, DPlot will draw the selected labels

without regard to whether the labels overlap. The "Other ASTM sizes" sieve labels will **only** be drawn if "Specify sieve size labels" is checked **and** "Use British standard sieve sizes" is **unchecked**.

#### Other sizes

If **Specify sieve size labels** is checked, you may enter up to 4 custom sieve sizes not shown in the list above. Sieve sizes with a 0 size value are ignored. The "description" label will be drawn at the top of the plot.

#### Use Udden-Wentworth scale rather than AASHTO scale

This setting effects the appearance of the aggregrate classification scale appearing below the X axis. For more information see Udden-Wentworth scale and/or AASHTO.

#### X Axis Units

**NOTE:** Changing units does **NOT** alter the input data in any way. By default, grain size distribution plots expect grain sizes in millimeters. This selection allows you to select inches or enter a conversion factor for user-defined units. If you select "user-defined" units, you <u>must</u> enter a positive value for the conversion factor. The grain size units are used to map sieve sizes to the X axis.

### 烤 Related macro commands

GrainSizeFlags GrainSizeFlagsEx GrainSizeOther

#### 4.4.1.5 Tripartite Options

This command allows you to change the appearance of tripartite grids.

## Shortcut:



Right-click within an existing tripartite plot and select Tripartite Options.

#### Units

**NOTE:** Changing units does **NOT** alter the input data in any way. Specifying alternate units informs DPlot how the input is related to certain aspects of the plot. It is up to you to perform any necessary conversions of the input data so that it matches the specified units. See Edit menu.

By default, a tripartite grid is set up for a shock spectra plot with frequency (X) in Hertz, velocity (Y) in inches/sec, displacement in inches, and acceleration in g's. This menu option allows you to select from other built-in relationships, or enter conversion factors for your own tripartite grids.

For shock spectra plots, acceleration and displacement values are given by:

```
d = v /2πf
a = -2πfv
where a = acceleration
d = displacement
v = velocity
```

In the more general case,

45° lines = A*Y/X 135° lines = B*X*Y where A and B are constants.

If you select "user-defined" tripartite units, you <u>must</u> enter positive values for the constants A & B. The following table is presented to provide examples of proper values for A and B:

Х	Y	45 deg.	135 deg.	A	В
Hertz	inches/ sec	inches	in/sec^2	0.159154 9	6.2831853

Hertz	inches/ sec	inches	g's	0.159154 9	1.6273972E- 02
Hertz	feet/sec	feet	g's	0.159154 9	1.9528766E- 01
Hertz	mm/sec	millimeters	g's	0.159154 9	6.4070659E- 04
Hertz	cm/sec	centimeter s	g's	0.159154 9	6.4070659E- 03
Hertz	meters/ sec	meters	g's	0.159154 9	6.4070659E- 01

If X represents the period of the oscillation rather than the frequency in cycles/second, the above relationships are simply reversed: The 45 degree lines represent acceleration and the 135 degree lines represent displacement:

Х	Y	45 deg.	135 deg.	A	В
second s	inches/ sec	g's	inches	1.6273972E- 02	0.159154 9
second s	feet/sec	g's	feet	1.9528766E- 01	0.159154 9
second s	mm/sec	g's	millimeters	6.4070659E- 04	0.159154 9
second s	cm/sec	g's	centimeter s	6.4070659E- 03	0.159154 9
second s	meters/ sec	g's	meters	6.4070659E- 01	0.159154 9

## Draw intermediate acceleration/displacement lines

By default, grid lines will be drawn for acceleration and displacement values of 1 and 5 times some power of 10, with only the power of 10 lines labeled. If this option is checked, the 0.5, 5., 50., etc. lines will also be labeled, and grid lines will also be drawn at 2, 3, 4, 6, 7, 8, and 9 times a power of 10.

# Related macro commands

**TripartiteFlags** 

# 4.4.2 Grid Lines or Box command

Controls the appearance of the X and Y axes.

Axes Only

A single line for each of the X and Y axes, with tick marks at appropriate intervals.

Grid Lines

Draw a box around the plot, with horizontal and vertical grid lines. Grid line color may be controlled with the <u>Colors</u> command on the Options menu and/or by clicking the Colors button on the toolbar. The width and style of the grid lines may be controlled with the <u>Symbol/Line Styles</u> command.



Draw a box around the plot, with tick mark intervals at the appropriate intervals.

# Vertical Grid Lines Only

Draw a box around the plot, with grid lines in the vertical direction and tick marks in the horizontal direction. As with **Grid Lines**, line color may be controlled with the <u>Colors</u> command on the Options menu and/or by clicking the Colors button on the toolbar. The width and style of the grid lines may be controlled with the <u>Symbol/Line Styles</u> command.

## Horizontal Grid Lines Only



Draw a box around the plot, with grid lines in the horizontal direction and tick marks in the vertical direction. As with **Grid Lines**, line color may be controlled with the <u>Colors</u> command on the Options menu and/or by clicking the Colors button on the toolbar. The width and style of the grid lines may be controlled with the <u>Symbol/Line Styles</u> command.

## Arrowheads

Toggles the drawing of arrowheads on the coordinate axes if Axes Only is checked.

Some scaling options take precedence over the grid type selection. See Linear/Log scaling



AxisArrowheads AxisType GridLines

# 4.4.3 Symbol/Line Styles command

You may select from 7 line patterns and/or 38 built-in symbol styles. You may also select any of 672 characters from the Wingdings fonts for use as symbols.

To create a scatter plot (symbols at each data point, no line segments connecting the points) select 'None' for the line style of each curve.

#### Line width

If any line width is set to 0.0 inches, DPlot uses one unit (one printer dot or one pixel) for that line width. For output to plotters, DPlot uses a single pen width regardless of the width you specify.

**NOTE:** Colors and/or gray shades may not be printed correctly with very small line widths if your printer must simulate colors with dithered patterns. For example, an HP LaserJet uses patterns of black and white dots to simulate various colors and gray shades. Similarly, many color printers simulate a wide range of colors by using patterns of cyan, magenta, and yellow dots. If the pattern used by the printer to produce these simulations is wider than the specified line width, the output will not be as you expected. In particular, a 0.0" line width (1 printer dot) may result in a single white (invisible) line. For best results when printing colors or gray shades, the minimum line widths for all curves and grid lines should be set to no less than 12 dots, or 0.02 inches on a 600 dpi printer or 0.04 inches on a 300 dpi printer.

## Symbol types

Select a symbol from the list, or click the "Other..." button to use a character from the Wingdings, Wingdings 2, or Wingdings 3 fonts. **Note:** These fonts must be installed on your system for this feature to work properly.

**NOTE:** There are two sets of inverted triangle symbols. The first two are placed such that the geometric center of the triangle is at the corresponding data point. The second two are placed such that they align vertically with the non-inverted (apex at the top) triangle symbols.

## Symbol size

If the symbol size is set to 0.0 inches, DPlot uses a default symbol height of 1/40 of the height of the

output device.

#### Cycle symbol colors

If checked, symbols in the selected curve cycle through the color palette rather than using the color associated with that curve. The length of the cycle is limited to the number of curves on the plot.

#### Symbol sizes in Y units

If checked, the symbol size uses the same scaling as the Y values on the plot. If you zoom in/out or change the extents of the plot, the symbols will be resized accordingly. This option applies to **all** symbols on the plot; you cannot have symbols for one curve be a fixed physical size and symbols on another curve proportional to the plot amplitudes. Symbols in the legend will be drawn a fixed size, roughly the same height as the accompanying text (if this option is unchecked then legend symbols will be the same size as those used in the plot). This option is disabled for logarithmic and probability scales on the Y axis, since there is not a 1:1 relationship between Y values and physical distance between those values.

## Symbol frequency

Enter the frequency at which symbols are drawn. For example, if set to 1 then symbols will be drawn at every data point; if 2 then symbols are drawn at every other point. This option might be useful for plots with a large number of data points.

## Symbol opacity

Enter the opacity of symbols as a percentage, from 0 (completely transparent: not drawn, in other words) to 100 (opaque). The opacity value is ignored for output to standard and enhanced metafiles and to displays for which the color depth is 256 or fewer colors. This option might be useful for depicting the density of data points in a plot. For opacity values less than 100%, the **Outline solid symbols** option is ignored.



You can quickly set the line width, symbol size, symbol frequency, and symbol opacity for ALL curves to the same values by selecting the **Apply below to all curves** checkbox.

#### Outline solid symbols

If checked, solid filled symbols (not including Wingdings characters) will be outlined in the color selected below this checkbox. If unchecked, symbols are drawn with the color associated with that curve. The checkbox setting is applied to all curves; the color selection is specific to each curve.

#### Fill hollow symbols with bg color

If checked, symbols with hollow interiors (not including Wingdings characters) will be filled with the background color of the plot (or legend, for symbols drawn in the legend).

■ Example:

Fill hollow symbols with bg color Fill hollow

Fill hollow symbols with bg color



## Shortcut:

Click on the toolbar.

or

or

Right-click on curves to select line and symbol styles and sizes.

Ctrl+click on a curve to bring up the Symbol/Line Styles dialog with the clicked-on curve selected.

## Selated macro commands

GridLinesOnTop GridLineType LineType LineWidth SymbolClip SymbolCycleColors SymbolFillBackground SymbolFrequency SymbolOpacity SymbolOpacity SymbolOutline SymbolOutline SymbolOutline SymbolSize SymbolSize SymbolSizeUnits SymbolType

## 4.4.4 Extents/Intervals /Size command

Allows you to select the extents of the plot, the number of tick marks and/or tick mark interval, and the physical size of the plot.

#### Specify extents

If checked, you may select the coordinates of the plot window corners. This is often useful if you are making plots of multiple data sets and want the extents of each plot to be identical. Use of this option may result in portions of the plot being clipped. For *Calendar Date* and *Date and Time* <u>number</u> formatting the extents may be entered in the date format selected with the <u>Number Format</u> command ^t, or as numbers using the Excel date system. For *Time* number formatting the extents may be entered in h:mm:ss format.

**NOTE:** The actual coordinates of the corners may not turn out to be exactly the same as those you have specified. DPlot will adjust the extents such that the resulting tick mark intervals are multiples of 1, 1.5, 2, 2.5, 3, 4, 5, 6, or 8 times a power of 10. If a tick mark interval or number of tick mark intervals is not specified (see below), these settings should be viewed as more of a suggestion than a firm setting, as DPlot will make a good tick mark interval the priority, and that interval may not necessarily divide evenly into your specified extents.

If unchecked, DPlot selects the coordinates for the plot window so that the entire plot will fit within the

window (before any zoom operation is performed).

#### Invert X, Invert Y (XY plots only)

If present, checking these boxes will invert the scaling on the corresponding axis, whether or not *Specify Extents* is used. For example on the X axis the "Low" value will be at the right end of the scale, the "High" value will be at the left end of the scale. This functionality does not apply to polar charts, tripartite grids, grain size distribution plots, or triangle plots. For 3D plots, you can accomplish the same thing by using negative scale factors on the <u>Contour Options</u> dialog box.

## **Exceptions**

If multiple independent Y axes are used with logarithmic scaling, the axis extents will be forced to the nearest power of 10 that includes the specified extents. For other scale types, the axis extents will be forced to the nearest number with two significant digits that includes the specified extents.

These options are disabled for <u>triangle plots</u>; the extents of all axes are fixed at 0-100 or 0-1, dependent on the <u>Triangle Plot Options</u> X+Y+Z=1 setting.

## Tick marks

## Auto-tick marks

DPlot automatically selects the number of tick marks and the interval between tick marks such that the interval is as described above.

## Specify number

If selected, you specify the number of tick marks on the axes. This option has no effect on the number of tick mark intervals for logarithmic or probability scales.

For polar coordinates, the number of tick intervals in the Y direction is used as the number of radial tick marks, and the number of tick intervals in the X direction is the number of angular divisions. You can force polar plots to be circular by specifying identical measurements for the width and height (see below), **or** by checking the "Force circular plots" option on the <u>Polar Plot</u> <u>Options</u> dialog.

### Specify interval

If selected, you specify the interval between tick marks. Again, this option has no effect on logarithmic or probability scales.

If necessary, the extents of the plot are adjusted to be multiples of the tick mark interval.

#### **Exceptions**

Tick mark settings are ignored for calendar dates if the extents of the applicable axis are greater than 60 days. If the extents are greater than 60 days but less than 120 days, tick marks are drawn at the 1st and 15th of every month. If greater than 120 days, tick marks are drawn at the 1st of every month only.

#### **Minor Grid/Tick Divisions**

If greater than 1, minor grid lines or tick marks are drawn between the major grid/tick marks with a pen width specified on the <u>Symbol/Line Styles</u> dialog. For logarithmic and probability scales the actual value is ignored; minor lines/marks are either on (>1) or off (=1). Depending on screen resolution, this option may result in the entire background of a plot being painted in the grid line color, but generally has a much sharper appearance on printed output.

For date scales on XY plots, the actual interval used will be dependent on the size of that interval (major interval divided by the number of minor intervals). In other words the actual number of intervals may not match the number requested. For minor intervals greater than 10 days and less than 5 years, DPlot will draw minor tick intervals at the 1st or 15th of each month for intervals less than 20 days, or the start of the next month, 2, 3, 4, or 6 months, or the next January 1 or the January 1 of the following year. For minor intervals greater than or equal to 5 years, DPlot will generally pick January 1 in whatever year comes closest to the specified interval. For minor intervals less or equal to 10 days the interval will be exactly equal to the major interval divided by the number of intervals, with no regard for aligning tick marks with whole days.

#### Specify plot size

If checked, you may set the physical size of the plot. For all plot types, these values specify the size of a box that will contain the plotted data, **not** including title lines, axis labels, or numbers along each axis. If unchecked, DPlot uses the full extents of the output device.

For output to printers and other hardcopy devices, these measurements are exact (to the resolution of the device). On a display monitor, the size of the plot is mapped to "logical inches". (Since displays are typically viewed from a greater distance than printed material, Windows scales font sizes so that characters with the same height appear roughly the same size whether viewed on a display monitor or a printout.) The result of this mapping is to provide a plot whose dimensions and font sizes appear to be the same size, whether viewed on the display or a printed copy.

You can also set the size of 2D plots by pointing the mouse cursor at a corner of the plot and dragging:



Dimensions will be shown in the units selected with the **Default dimensions** setting on the <u>General</u> <u>command</u> of the Options menu.

**NOTE:** If the specified size exceeds the capabilities of the output device (subtracting the margins set with the Page Setup command on the File menu), DPlot adjusts the size to one that will fit, maintaining the aspect ratio between the width and height.

For polar plots with the <u>"Force circular plots"</u> option in effect, DPlot will override the width and height entries if necessary, using the smaller of the two specified dimensions.

#### Set X:Y=1:1

This button is disabled unless both *Specify Extents* and *Specify Size* are checked. When pushed, the size of the plot is adjusted such that the physical scale of the X and Y values are equal: 1 unit in X will be the same length as 1 unit in Y. This action is not persistent: if you subsequently edit either the extents or the size then this relationship will no longer be true until you push the button again.

[†] Date format: Calendar dates will initially be presented in the date format selected with the <u>Number</u> <u>Format</u> command. Calendar dates may be entered in any variation on the current date format selected with the <u>Number Format</u> command, as long as the day, month, and year order is preserved. For example if the date format is M/d/yyyy then any of M/d/yy, MM/dd/yy, M/dd/yyyy, etc. will be interpreted correctly. If the date format under the Number Format command is set to "<none>", then the initial display and subsequent interpretation is left up to the "Assume input dates are of the form d/m/y" setting of the <u>General Options</u> menu command. (If unchecked, calendar dates are assumed to be in the order m/d/y.)

## 差 Related macro commands

AutoScale DenseGridLines InvertAxis ManualScale NumTicks Size TickInterval
See also: Fonts

# 4.4.5 Square log cycles command

This option forces the physical distance between log cycles to be the same on the X and Y axes. It has no meaning for scaling options other than Logarithmic X-Logarithmic Y or Tripartite grids. The "Set Plot Size" option of the <u>Extents/Intervals /Size</u> command precedence over this option and, if checked, causes this menu command to be disabled.

# 烤 Related macro commands

<u>SquareLogCycles</u>

# 4.4.6 Log scale tick labels command

These settings allow you to determine the frequency of tick mark labels on logarithmic scales. In **all** cases, DPlot will **not** draw a label if doing so would overlap a label with a higher precedence. Label precedence is: 1, 5, 2 and 7, 3 and 6, 4 and 8, 9 times the respective power of 10. For example, the label '2' will be drawn only if it will fit between the '1' and '5' labels; '6' will be drawn only if it will fit between the '5' and '7'.

### Powers of 10 only

Draws only powers of 10 (0.1, 1.0, 10, 100, etc.) plus the axis extents.

### **Fives**

Also draws labels at tick marks equal to 5 times a power of 10.

#### All major tick marks

Draws labels at all major tick marks.

#### Minor tick marks

If minor tick marks are drawn, also draws labels at those tick marks

# 烤 Related macro commands

**LogTicks** 

# 4.4.7 Multiple Axes command

Displays a dialog box that allows you to specify the appearance of the coordinate axes.

# $X2 = A+B^*X1$

A second X axis is linked to the primary X axis by the function  $X2 = A + B^*X1$ . This option might be useful for displaying output in two different units, for example. You can enter equations from either coefficient.

■ Example

These settings:

KII Multiple Axes	X
X Axis <u>O</u> ptions: Single X axis X2 = A + B <b>*</b> X1	∆: 0 <u>B</u> : pi/180
Y Axis Options:	
For grid lines, use ind	ependent tick marks on dependent scales Cancel <u>H</u> elp

...along with "Pi Multiples" for the number format on the second X axis results in:





useful for displaying output in two different units, for example. Note that the function relating Y2 to Y1 is a relationship between the numbers, not the units, displayed alongside the axes. For example, if the primary Y axis has units of inches, and the second Y axis has units of centimeters, then C=0, D=2.54 (not 1 cm = 0. + 0.3937 inches). As with X, equations may be used for C and D.

#### **Independent Y2**

A second Y axis is independent of the primary Y axis. In this case, you specify which curves are associated with which axis. The first curve in the plot **must** always be associated with the primary Y axis. This option might be useful for plotting, for example, pressure-time histories, with pressure associated with the primary Y axis and impulse associated with the second Y axis. Since the first curve must always be associated with the primary Y axis, this option is disabled if the plot has only one curve. For an example of this setting in action see <u>EX04.GRF</u> - <u>Multiple independent Y scales</u>.

## For grid lines, use independent tick marks on dependent scales

If checked and <u>grid lines</u> are used, DPlot will draw labels on the dependent scales at "nice" intervals, identical to those drawn if *Box Around Curves* is used instead of *Grid Lines*. With X2=0.3048*X1, Y2=2.54*Y1:



If unchecked, labels will be drawn at the grid line interval, with no additional tick marks:



This setting is applicable to linear and logarithmic scales.

# 💋 Related macro commands

IndependentTicks MultipleXAxes MultipleYAxes

# 4.4.8 Fill Between Curves command

This command allows you to draw a hatch pattern or solid fill:

- a) between two curves.
- b) between a curve and an edge of the graph or a horizontal line with specified Y value.
- c) below the curve to the bottom edge of the graph, only for points that at less than or equal to a specified Y value.
- d) that fills an area formed by closing a single curve by connecting the first and last points.

Select the curves to fill between in the <u>1</u>st Curve and <u>2nd Curve</u> list boxes. If **None** is selected for the 1st curve, the fill operation is not performed. To fill an area formed by closing a single curve, select **None** in the <u>2nd Curve</u> list box.

The color of the fill may be controlled by clicking on the **Color** control. Click on one of the 40 predefined colors to select it, or click on **More Colors...** to specify another color. If *Fill to* Y= is selected in the **2nd Curve** list box, then 1) the primary color becomes optional; if *Color Above Yref* is not checked then the color control will be hidden; 2) an optional *Color Below Yref* checkbox is displayed, and similarly, if checked then a second color control is shown that allows you to use a different color for areas below the reference Y value. If when you click OK both *Color Above Yref* and *Color Below Yref* are unchecked, the fill will be discarded.

If the **User-defined** fill pattern is selected, you may create your own hatch pattern by clicking on any of the grid squares in the 8x8 grid pattern. Clicking on a square toggles its state to and from transparent and the color selected with the **Color** control.

# Intersecting Fills

If the *Draw only the intersections of all areas* box is checked and more than one fill area is specified, then only the overlapping region for all fill areas will be drawn. In this case the color and fill style for the first fill area will be used; all other colors and fill styles are ignored. DPlot will not issue a warning if there is no overlapping area; it will simply not draw any fill. This feature is particularly useful for finding the solution to a set of inequalities.

Without Draw only the intersections of all areas checked:



With Draw only the intersections of all areas checked:



# ■ Effect of Amplitude Limits

When filling a single curve (**2nd Curve** set to **None**), if the <u>Amplitude Limits</u> command on the Options menu has been used to specify a range in Y outside of which points will be ignored, then the fill

region will be broken up into multiple segments if any points in the **<u>1</u>st Curve** are outside those limits. For example, in this map of Alaska all data points are in the same curve. The file import plugin used to read this file has inserted intermediate points between land masses with Y values outside the range that the plugin has specified for the amplitude limits (in this case +/- 90 degrees).



### Drawing Operation

Colors may be drawn with a simple paint operation (Copy, in which case the final color will be the pattern color) or using a mask operation (Mask), such that the result will be a color with r,g,b components common to both the pattern and the destination. For example, magenta (0x00FF00FF) on cyan (0x0000FFFF) results in blue (0x000000FF). In all cases, fill patterns are drawn **before** all curves; patterns will not obscure a curve nor will fill colors be combined in any way with curve colors.



#### Legend formatting

- To use subscripts, select the text (highlight using the cursor) you want to format as subscript, then click this button. **Shortcut:**  $CTRL+\downarrow$
- To use superscripts, select the text (highlight using the cursor) you want to format as superscript, then click this button. Shortcut: CTRL+
- To insert special characters (Greek letters, trademark symbols) found in the Symbol font, click on the insertion point, then click this button. Shortcut: CTRL+S

**NOTE:** On versions of Windows older than Vista, the Symbol font with normal weight may only be available in a horizontal orientation. Bold weight can generally be rotated to any angle. This limitation is of course important for the Y axis label on all plot types, and all axis labels in 3D views.

- **B** To make a portion of the text **bold**, select the text (highlight using the cursor) you want to format as bold, then click this button. **Shortcut:** CTRL+B
- To make a portion of the text *italic*, select the text (highlight using the cursor) you want to format as italic, then click this button. **Shortcut:** CTRL+I
- To make a portion of the text <u>underlined</u>, select the text (highlight using the cursor) you want to format as underlined, then click this button. **Shortcut:** CTRL+U
- **t2** To double the size of one or more characters, select the applicable characters (highlight using the cursor), then click this button. **Shortcut:** CTRL+2
- To use a custom color on a portion of the text, select the text (highlight using the cursor) you want to draw using a different color, then click this button. **Shortcut:** CTRL+K

#### Format codes

The formatting buttons simply insert special characters into the text that have specific meaning to DPlot. Use of these buttons is not strictly necessary - you can of course physically type these codes with the same result. Format codes may be nested, so you can have (for example), subscripted Greek characters or subscripts of subscripts. Note that the bold, italic, and underline codes will have no effect if the font already possesses that attribute. In other words, using {\b} on a bold font will **not** make the font "bolder".

Format	Code	Example	Result
Bold	{\btext}	{\bBold} text	Bold text
Italic	{\itext}	{\iltalic} text	Italic text
Subscript	{\dtext}	P{\dso}	P _{so}
Superscript	{\utext}	X{\u2} + Y{\u2}	$X^2 + Y^2$
Symbol	{\stext}	e{\u(-{\sa}t)}	e ^(-at)
Underline	{_text}	{_underlined} text	underlined text
Enlarged	{\2 <i>text</i> }	{\2{\sS}}x	Σx
Color	{\cBBGGRR <i>text</i> }	{\c0000FFRed} Text	Red Text
Overbar	{\otext}	{\oY}=56.23	<u></u> ¥=56.23
Dot accent	{\. <i>text</i> }	$Flow = \{ \backslash. V \}$	Flow = V
Double dot accent	{\: <i>text</i> }	$y = y{\d0} + {\.y}t - {\.y}t{\u2}/2$	$y = y_0 + \dot{y}t - \ddot{y}t^2/2$
Wingdings	<b>{\w1</b> <i>text</i> }	{\w1N} Danger {\w1N}	≵ Danger ≵
Wingdings 2	<b>{\w2</b> text <b>}</b>	{\w2R} Passed	☑ Passed
Wingdings 3	{\w3text}	Arrows {\w3!"#\$} and\nmore arrows {\w3NOPQ}	Arrows ←→†∔ and more arrows ⊊ ⊃೦೦

# Related macro commands

**FillBetween** 

# 4.4.9 Number Format command

# Shortcut:

Right-click on numbers along any axis to select the number format for that axis.

Displays a dialog box that allows you to specify the appearance of numbers on the axes of the plot. The following table shows some examples using the available number formats.

With this format	This number	Is displayed as
Default	-1.5	-1.5
	0.	0.
	1.5	1.5
None	any	nothing
Scientific	-1.5	-1.5E+0
	0.	0E+0
	1.5	1.5E+0
Powers of 10	150	1.5x10 ²
	0	0
	0.015	1.5x10 ⁻²
Engineering		identical to Powers of 10, but all exponents are multiples of 3
Percent	1.5	150%
Currency 1	-1.5	(\$1.50)
see below	0.	\$0.00
	1.5	\$1.50
Currency 2	-1.5	(\$2)
see below	0.	\$0
	1.5	\$2
Time of day (hours)	1	*01:00:00 AM
	1.5	*01:30:00 AM
	13	*01:00:00 PM
		or 13:00:00
Time of day (fractional days)	0.04166667 (1/24)	*01:00:00 AM
	0.25	*06:00:00 AM
	0.5	*12:00:00 PM
Calendar date	1	*1/1/1900

	35242	*6/26/1996
Date and Time	35242	*6/26/1996 12:00:00 AM
	35242. 5210069	*6/26/1996 12:30:15 PM
Day of the Week	1	Sun
	2	Mon
	8	Sun
Month	1	Jan
	12	Dec
Elapsed time	-1.5	-1:30:00
	0	0
	1.5	1:30:00
Fractions	0.4	² / ₅
	1.5	<b>1-</b> ¹ / ₂
Fractions, no reduction see below	0.4	⁴ / ₁₀
Pi Multiples	0.785398	0.25
	4.71239	1.5
Pi Fractions	0.785398	/4
	4.71239	3 /2
Degrees, Minutes	35.505556	35° 30.3'
Degrees, Minutes, Seconds	35.505556	35° 30' 20"
Hexadecimal	5346.24	0x14E2

* dependent on Date Format and/or Time Format settings

#### Axis

The X2 and Y2 axis entries (XY plots only) correspond to the secondary axes established by the <u>Multiple Axes</u> menu command. These selections are disabled if the corresponding axis has not been enabled. The Orientation settings are disabled for these entries and are forced to the orientation setting for the corresponding primary axis.

# Format

#### **Default format**

Numbers with absolute values smaller than 0.00001 or larger than or equal to 10,000,000 are displayed in scientific notation with up to 5 significant figures. Trailing zeros are not displayed.

# None

This format is primarily useful for creating blank graph paper. See the <u>How do I?</u> topic for more information.

# Fractions

DPlot will pick the denominator from 2,3,4,5,6,8,10,12,15,16,20,24,25,30,32,40,50,60,64 that gives the smallest error. Fractions smaller than 1/128 are truncated. The **Fraction** format will reduce the fraction to its lowest terms, e.g. 3/60 becomes 1/20. **Fractions, no reduction** is identical other than that reduction: if the tick mark interval on an axis is 1/60 and a tick mark is at 3/60, it will be displayed as 3/60 rather than 1/20.

### Currency

DPlot uses the currency format set with Control Panel's Regional Settings applet.

#### Time of day

For "Time of day (hours)", amplitudes should be in hours. Note that this differs from "Date and Time" and "Time of day (fractional days)" formats, in which time values are fractions of days. The time scale wraps around every 24 hours, with 0.0 equal to midnight (-1.0 = 23.0 = 47.0 = 11:00 PM, etc.) By default, DPlot uses the time format set with Control Panel's International (or Regional Settings) applet. You can control how the time is formatted by entering a format string in the Time Format box.

Time format notation:

h = hour, m=minute, s=second, t=AM or PM (ignored for 24 hour clock)

h = 12 hour clock, H = 24 hour clock

hh or HH = leading zero on hour

h or H = no leading zero on hour

A '.0' suffix will add tenths of seconds, '.00' will add hundredths of seconds, etc. You can display time increments as small as microseconds with '.000000'. A seventh or subsequent 0 will be displayed but is ignored for formatting purposes.

#### Calendar date

DPlot converts the input Excel serial number to a calendar date. By default, DPlot uses the Short Date format specified in Control Panel's International (or Regional Settings) applet for formatting the date. You can control how the date is formatted by entering a format string in the Date Format box. Use the letters in the following table to change the date style. As you type, the current date formatted using this style is shown below the style box.

To display	Use
Numbers with no leading zeros for single-digit days, years, or months.	Lowercase <b>d</b> or <b>y</b> , or uppercase <b>M</b>
Numbers with leading zeros for single-digit days, years, or months.	Lowercase <b>dd</b> or <b>yy</b> , or uppercase <b>MM</b>
Three-letter abbreviations for the day of the week or month.	Lowercase <b>ddd</b> or uppercase <b>MMM</b>
The full name of the day of the week, year, or month	Lowercase <b>ddd</b> or <b>yyyy</b> , or uppercase <b>MMMM</b>
The day of the year (Jan 1=1, Dec 31=365 or 366)	Lowercase <b>n</b>

#### **Date and Time**

Identical to **Calendar date**, with the exception that **Calendar Date** will sometimes include the time depending on the spacing between data values and the context, while **Date and Time** will always show the time in addition to the date, dependent on the **Time format** string. If you want to show **only** the time using this format, use **<none>** for the date format string.

#### Hexadecimal

Displays numbers in signed hexadecimal notation, dependent on the *Size* and *Prefix* settings under *Hexadecimal Parameters*. Fractional values are truncated. Numbers outside the range -0x80000000 to 0x7FFFFFFF (-2,147,483,648 to 2,147,483,647) are displayed using the default numerical format. The *Size* parameter will be ignored for amplitudes greater than the maximum value for that size. For example, if you select *1 Byte* and a value is outside the range -128 to 127, *2 Byte* will be forced on.

Degrees, Minutes and Degrees, Minutes, Seconds

The Use N/S/E/W suffix setting is only applicable to these formats. This setting is only used for numbers displayed along the X or Y coordinate axes for linear scales or Mercator projections. (It is **not** used, for example, in dialog boxes, the status bar, popup windows displaying coordinate values, or any other scale type.) If checked, the absolute values will be displayed along with the appropriate suffix (N or E for positive values, S or W for negative values).

#### Use 1000 Separator (,)

Check this box to display the thousand's separator character before every 3rd digit in the whole number portion of a value. The separator character is set by Control Panel's Regional Settings applet.

#### **Decimal places**

If *Fixed number* is checked, then the corresponding number of figures (including trailing 0's) are displayed after the decimal point. This option is only used for the default number format and for percentages, and is ignored for absolute values less than 0.00001 or larger than or equal to 10,000,000. **Please note** that this option does not enforce settings that make sense: 0.0001 will be displayed as 0.000 if 3 digits are selected, for example.

#### Orientation

Allows you to control the orientation of the numbers along each axis (default is horizontal). This option is ignored for polar plots, triangle plots, and 3D plots, and for secondary axes on XY plots is forced to the setting for the corresponding primary axis.

**Note:** Most moderate-sized non-bold fonts will have a pronounced aliasing problem at orientations other than horizontal or vertical on your display. They're ugly, in other words. The equivalent **bold** font will generally have a better appearance in this case. This applies mainly to the display and saved bitmaps; printed fonts will generally look good regardless of the orientation or font weight.

# Selated macro commands

DateFormat DecimalDigits LabelDirection NSEW NumberFormat SplitDateTime TimeFormat

# 4.4.10 Colors command

Allows you to specify the colors that will be used for drawing lines/curves, grid lines, and the coordinate axes on an XY plot.

Edit

Presents the Colors dialog box which may be used to alter the color palette.

#### Shortcut:

Click	on the toolbar.

or

Shift+click on a curve to bring up the Colors dialog with the clicked-on curve selected.

### Colors dialog settings

## Interpolate between...

Generates a smooth blend of colors between the starting and ending indices, interpolating the red, green, and blue components as well as the luminance.

#### Recycle color values every <N> colors

Copies the color values of the first N colors to all subsequent groups of N colors. For example with N=10, colors 11, 21, 31,... will match color 1, colors 12, 22, 32,... will match color 2, etc.

Please note: Both the Interpolate between and Recycle functions are applied only when you

click the corresponding **Go** button. If you make subsequent changes to any color values then these settings are not automatically re-applied.

### Make defaults

Stores the current color settings so that these colors will be used as the defaults for all new plots.

# **Restore Defaults**

Restores the default color specifications. When DPlot is first installed, the default colors are:

Index	R	G	В	
Axes	0	0	0	
Grid '	192	192	192	
Curve 12	255	0	0	
22	255	0	85	
32	255	0	170	
42	255	0	255	
5	170	0	255	
6	85	0	255	
7	0	0	255	
8	0	85	255	
9	0	170	255	
10	0	255	255	
11	0	255	85	
12	0	255	0	
13 ⁻	170	255	0	
142	255	255	0	
152	255	170	0	
16 3	255	85	0	

The default colors for subsequent curves recycle the above 16 colors. For example the default color for curve #17 is identical to the color for curve #1.

The default color palette may be changed by checking the **Make defaults** box on the **Colors** dialog. **Print B&W** 

Ignores color palette selections and prints all curves and grid lines in black.

## Related macro commands

AxisColor BkColor BkPlotColor Color ColorCycle

# 4.4.11 Amplitude Limits command

This dialog allows you to specify limits outside which data points will be ignored. Line segments will not be drawn to or from these data points, nor will line segments be drawn between the last and next data points on either side of an ignored point. This allows you to create gaps in your curves. There are several potential uses for this feature; here is one:

The plot below was created using  $\underline{Y=f(X)}$  with Y=tan(X) from PI/4 to 3*PI/4, then zooming in to the limits shown. The only problem with this plot is the vertical line segment between just before X=PI/2 to just after X=PI/2, which has no real meaning.



To eliminate this vertical line, we use Amplitude Limits and set the limits to +/- 200:



As another example, some data sources use amplitudes well outside the limits of any expected results for missing data. For example, you can get temperature data for many locations from <a href="http://data.giss.nasa.gov/gistemp/station_data/">http://data.giss.nasa.gov/gistemp/station_data/</a>. Copying and pasting the temperature data for Port Gibson, Mississippi results in this not especially helpful graph:



The 999.9 spikes are missing data (temperatures for months with no data are recorded as 999.9, which is hopefully well outside the expected temperature in Mississippi!). You can of course zoom in to the real temperatures, but in this case it would likely be better to simply hide those "no data" points. To hide those points and get a more reasonable plot, you can use Amplitude Limits with the low value set to (in this case) 0 or less and the high value set to 30 or more:



Note that this differs from using <u>Clip Y</u> on the Edit menu, which would delete points outside a specified range but then connect the adjacent points. *Amplitude Limits* makes gaps in the data much more obvious.

### Related macro commands

YCutoff

# 4.4.12 Data Processing command

This command allows you to enter a data processing mode in which several operations are automated. The key features of data processing mode are geared towards producing data plots used by specific organizations and may not be appropriate for your use.

# Formatting

- Square format:
- Plot width and height are 5.15 inches (about 131 mm).
- Both the X and Y axes have 10 grid intervals.
- Wide format:
- Plot width is 6.15 inches; height is 3.69 inches.
- X axis has 20 grid intervals; Y axis has 12 grid intervals.
- Scaling is linear X, linear Y.
- Pen width is 1 dot or pixel for curves, 0.02 inches for the box around the plot, and 1 dot or pixel for the grid lines.
- All labels use the Arial TrueType font. Normal text is 10pt; the first title line is 18pt; all other text is 14pt.

### Units

Input units should correspond to the following table. If the input units do not match the values specified, you should **not** select "Use DPlot-generated axis labels".

Data type	Input X	Input Y	Output X	Output Y default
				<u>units</u>
Unknown	don't care	don't care	same	same
Pressure	Time,	Pressure, psi	Time,	Pressure, psi
	msec		msec	Impulse, psi-sec
Stress	Time,	Stress, psi	Time,	Stress, psi
	msec		msec	Impulse, psi-sec
Acceleratio	oTime,	Acceleration,	Time,	Acceleration,
n	msec	g's, feet/sec^2,	msec	same units as
		inches/sec^2,		input
		or meters/sec^2	2	Velocity, feet/sec
				Displacement,
				inches
Deflection	Time,	Deflection,	Time,	Deflection,
	msec	inches	msec	inches
Strain	Time,	Strain	Time,	Strain
	msec		msec	
Temperatu	rTime,	Temp., deg. F	Time,	Temperature,
е	msec		msec	deg. F

For acceleration plots you may specify the units of the input acceleration as well as the generated velocity and displacement curves. Output acceleration units must match the input units. For other data types the input units are only critical to the axis labels produced by DPlot if "Use DPlot-generated axis labels" is checked, since any integrations performed are simply the input values times time. For acceleration records, however, the unit selection must be correct, whether "Use DPlot-generated axis labels" is checked or not.

Y axis labels will be the same color as the corresponding curve, regardless of the color chosen with the <u>Fonts</u> command.

# Menu Commands

Several menu commands are disabled when in data processing mode:

- Erase a curve.
- Swap X,Y.
- Linear/Log scaling.
- Multiple axes.

- · Compare curves.
- Curve fit.
- Integrate a curve.
- Lump two curves.

In addition, the decimation commands do **not** generate a new curve ... they replace the curve that is operated on.

#### Integrations

For records consisting of pressure or stress values, DPlot automatically generates an impulse record. For acceleration records, DPlot automatically generates velocity and displacement records.

Editing the primary record in any way (pressure, stress, or acceleration) results in automatic regeneration of the integrated records. **Please note:** The initial amplitude of an integrated record will **always** be zero. Some operations on the primary record may have an undesirable or unexpected effect on integrated records. For example, truncating a pressure record (deleting data before a specified time), may result in shifting the impulse. To view or print a range in time without changing the integrated amplitudes, zoom in to the desired limits or use the <u>Extents/Intervals /Size</u> command rather than truncating the primary record.

Generated records (impulse, velocity, or displacement) can not be edited directly. Any changes must be made to the primary record.

### Acceleration plots

Acceleration is plotted on a separate plot from velocity and displacement. To switch views from acceleration to velocity, select the Velocity/Displacement command on the View menu (or press Shift+TAB). Switching to the velocity plot changes this menu command to "Acceleration".

To save a single curve, the curve must be visible. For example, to save either the velocity or displacement records using the <u>Save As</u> command, switch views if necessary so that the velocity plot is visible.

#### Array sizes

Since data processing mode relies on DPlot being able to make very specific assumptions about the content of the curves, you cannot change the maximum number of curves while in data processing mode. Doing so turns data processing mode off.

# 4.4.13 Bubble Plot command

In this context a bubble plot lets you chart three variables in two dimensions by varying the size and/or color of each data point in a scatter plot based on a third variable. In DPlot, that third variable is taken from the Y values of a separate curve. The source curve for the bubble amplitudes will not be drawn. DPlot ignores the X values for the source curve(s); the n'th point of the source curve is used as the bubble amplitude for the n'th point of the selected curve, regardless of whether the X values match up. This command is disabled if only 1 curve is present in the plot.

If you want to use the amplitudes of a given curve for the bubble values for that same curve, you can do this by making a copy of that curve. Select Edit>Copy>Data Values, select the curve (if more than one curve is present), then Edit>Paste. For the **Bubble amplitudes from** curve, select the copy.

The low and high symbol sizes and colors are mapped to the low and high amplitudes from the source curve, respectively. If you want all symbols to be the same size, enter identical values for the low and high sizes. Likewise if you want all symbols to be the same color, select identical colors.

You can control the symbol type and colors used by right-clicking on any of the symbols in a bubble plot.

Line segments are not drawn between data points for bubble plots, though you can mix curves using bubbles with curves that do not use this feature.

#### Example



# Earthquake Magnitudes, Jan 1, 1999 - Jan 7, 2001

For more information on how this plot was created, see the Maps page on the DPlot web site.

# 💋 Related macro commands

### **Bubbles**

# 4.4.14 Vector Plot command

A vector plot allows you to chart four variables in two dimensions by varying the lengths and angles of vectors drawn from each data point in a scatter plot based on a third and fourth variable. In DPlot, the third and fourth dimensions (angle and magnitude) are taken from the Y values of two other curves. The source curves for the vector angles and magnitudes will not be drawn. DPlot ignores the X values for the source curve(s); e.g. the n'th point of the angle source curve is used as the angle for n'th point of the selected curve, regardless of whether the X values match up. This command is disabled if fewer than 3 curves are present in the plot.

The low and high vector lengths and colors are mapped to the low and high absolute value of the amplitudes from the magnitude source curve, respectively. If you want all vectors to be the same size, enter identical values for the low and high sizes. Likewise if you want all vectors to be the same color, select identical colors.

Line segments are not drawn between data points for vector plots, though you can mix curves using vectors with curves that do not use this feature.

# Example



# Air Speed/Ground Speed

### Discrete colors (default = blend)

By default colors are blended between the low and high values (or between each color range if more than 2 colors are specified).

#### Show color key

If checked, DPlot generates a legend with optional heading. This color key may be moved around on the graph by clicking and dragging with your mouse. Attributes (framed, opaque, etc.) may be changed by right-clicking on the key or by double-clicking. The color key text is fixed and cannot be changed.

#### Specify magnitudes for colors

If unchecked, each color range is divided evenly between the low and high magnitude values. If checked, you may enter specific values for each color.

# Color key heading

Enter a heading for the color key. This entry is meaningless if **Show color key** is unchecked. For multiple lines, use \n sequences. (For C programmers, that is the actual characters \n, not a line feed). For example the text "First line\nSecond line" will be split into two lines as you'd expect.

# Example (Discrete (6) colors and Show color key checked)



Air Speed/Ground Speed

Related macro commands

VectorPlot VectorPlotColors VectorPlotOptions

# 4.4.15 OHLC Chart command

An open-high-low-close (OHLC) chart is typically used to show variations in stock prices. Each vertical line on the chart shows the price range (the highest and lowest prices) over one unit of time (normally one day). Tick marks project from each side of the line indicating the opening price on the left, and the closing price for the time period on the right. In DPlot, the high, low, and close values are taken from the Y values of three other curves. DPlot ignores the X (normally date) values for the source curves; e. g. the n'th point of the high source curve is used as the high value for n'th point of the selected curve, regardless of whether the X values match up. This command is disabled if fewer than 4 curves are present in the plot.

Individual entries may be drawn in the <u>color specified for the given curve</u>, or may be based upon whether the close price is greater than or equal to the open price or vice versa.

Tick mark length is always 0.5 in data units (1/2 day).

### Example:

Retrieve stock market data. This example uses data from <a href="http://finance.yahoo.com/">http://finance.yahoo.com/</a>, though there are of course others. In this case we'll import data for Google (symbol=GOOG). Click "Historical Prices", then scroll to the bottom of the page and click "Download to Spreadsheet". If given the option by your browser, save the file to disk rather than opening it. Open this file in DPlot. If you have associated CSV files with DPlot, right-click on the file in Explorer, select "Open with", then DPlot. You can of course also open the file by starting DPlot and selecting File>Open, file type D, then the file. Or you can open DPlot then drag the file to an empty DPlot document window (first select File>New if the active document contains a plot).

Deleting or not importing the **Volume** and **Adj Close** columns you should see a plot similar to this after zooming in to the year 2012:



Select "OHLC Chart" on the Options menu and make these selections:

Open-High-Low-Close		X
Open-High-Low-Close Open:	High: None 1: Open 2: High 3: Low Close: 1: Open 2: High 3: Low	OK Cancel Help
Close ≤ Open Close < Open Close < Open	Candlestick chart	

These selections should result in a plot similar to this:



You'll note that the Open and Close tick marks aren't visible in this case. That's due to a combination of the <u>line width</u> used and the <u>extents of the plot</u>. Tick marks are always drawn 1/2 unit (normally 1/2 day) long, so if the days are spaced too closely **and** the line width is too large, the tick marks won't be visible. If you zoom in or use <u>Options>Extents/Intervals/Size</u> to isolate on the last 2 months of 2012, you should get something similar to this:



Tick mark length (and visibility) are a function of line width, extents, and plot size.

Note that the legend entry still refers to the first curve rather than the data source. You can easily change the legend entry (if desired) by double-clicking on it or by selecting the <u>Legend/Labels</u> <u>command</u> on the Text menu. You can also hide the legend by right-clicking on it and selecting "Hide Legend".

If instead you would prefer a <u>candlestick chart</u>, select OHLC chart on the Options menu and check the "Candlestick chart" box. The last month of 2012 with the above data will look something like this:



Hollow boxes indicate a closing price greater than or equal to the opening price. Conversely, solid-filled boxes indicate an opening price greater than the closing price. In DPlot, the box width is always 2/3 of an X unit - 2/3 of a day in this example.

If you are curious about the date scale, this plot uses "ddd M/d" date format for the <u>Number Format</u> on the X axis.

# Related macro commands

**OHLC** 

# 4.4.16 Error Bars command

This dialog allows you to specify parameters for vertical and/or horizontal error bars for one or more curves.

Display

	•	
F	T	1

Error bars are not drawn, regardless of the settings under Error Amount.

Positive error bars extending up (or to the right, depending on the "Horizontal bars" setting) from each data point are drawn, subject to the conditions given by *Error Amount*.



Negative error bars extending down (or to the left) from each data point are drawn.

Both positive and negative error bars extending from each data point are drawn.

Error Amount	
Fixed value	Error bars for every data point have the same constant magnitude for both the positive and negative errors (subject to the <i>Display</i> setting).
Percentage	Error bar magnitude is a percentage of the amplitude of the respective data point.
Custom	Error bars use different positive and negative values. Magnitude of the error bar is constant for all data points.
Different error values for every point	Error values are retrieved from another curve or from two curves. Curves that are selected to provide error values will not be drawn nor will they be considered when finding the plot extents. DPlot ignores the X values for these curves; the n'th point of the <i>Plus</i> curve is used as the error value for the n'th point of the selected curve, regardless of whether the x values match up. Both the + and - error curves should consist of <b>positive</b> magnitudes. This option is disabled if only 1 curve is present in the plot.

**NOTE:** Error bars on bar charts will always be drawn parallel to the bars, regardless of whether you specify Vertical or Horizontal error bars.

# Related macro commands

ErrorBar ErrorBarPenWidth ErrorBarWidth

# 4.4.17 Reference Lines command

Displays a dialog box that allows you to specify up to 50 horizontal and/or vertical lines on an XY plot or a 2D view of 3D data at specified constant X or Y values, or a circle with a constant radius or radial line at a fixed angle on a polar plot, or lines on a 1D plot as specified Y values. Each line may have a distinct color, width, and style.

You can click on the plot to select the location of a reference line. This feature is enabled when any control within the "At Value" box has the keyboard focus.

You can also use <u>text placeholders</u> for the location of the reference line. For example, with "Y=" selected if you enter "=\$YMAX" then the reference line will be drawn at the maximum Y value; with "X=" selected if you enter "=\$XYMAX" then the line will be placed at the X coordinate where the maximum Y value occurs. (If you use text placeholders or any other equation in this way, the expression is evaluated and translated to a number when you click OK. In other words, with "=\$YMAX" as the

reference line value, the line will not shift if the Y values are changed in any way. This differs from using text placeholders in a text entry like title lines or the legend, where the evaluation takes place at the time the plot is rendered.)

The **Frame** and **Opaque** checkboxes are only relevant if a **Label** is entered and "Draw label on plot..." is checked. "Draw label on plot..." is not applicable to 2D views of 3D data, for which the label will **always** be drawn centered on the reference line.

Check **Hide** to hide a specific reference line (but not delete it). This option might be useful if you have more than a few reference lines and want to hide several for a specific application but not delete them.

#### Label

Text associated with the line that will appear in the plot legend, or centered on the line if "Draw label on plot..." is checked. If you do not want a label entry for the line, leave this entry blank. This entry may include the same special formatting codes used in the legend and title lines:

- $x_2$  To use subscripts, select the text (highlight using the cursor) you want to format as subscript, then click this button. **Shortcut:** CTRL+ $\downarrow$
- **x**² To use superscripts, select the text (highlight using the cursor) you want to format as superscript, then click this button. **Shortcut:** CTRL+
- To insert special characters (Greek letters, trademark symbols) found in the Symbol font, click on the insertion point, then click this button.
   Shortcut: CTRL+S

**NOTE:** On versions of Windows older than Vista, the Symbol font with normal weight may only be available in a horizontal orientation. Bold weight can generally be rotated to any angle. This limitation is of course important for the Y axis label on all plot types, and all axis labels in 3D views.

- **B** To make a portion of the text **bold**, select the text (highlight using the cursor) you want to format as bold, then click this button. **Shortcut:** CTRL+B
- To make a portion of the text *italic*, select the text (highlight using the cursor) you want to format as italic, then click this button. **Shortcut:** CTRL+I
- To make a portion of the text <u>underlined</u>, select the text (highlight using the cursor) you want to format as underlined, then click this button. **Shortcut:** CTRL+U
- **t2** To double the size of one or more characters, select the applicable characters (highlight using the cursor), then click this button. **Shortcut:** CTRL+2
- To use a custom color on a portion of the text, select the text (highlight using the cursor) you want to draw using a different color, then click this button. **Shortcut:** CTRL+K

### Format codes

The formatting buttons simply insert special characters into the text that have specific meaning to DPlot. Use of these buttons is not strictly necessary - you can of course physically type these codes with the same result. Format codes may be nested, so you can have (for example), subscripted Greek characters or subscripts of subscripts. Note that the bold, italic, and underline codes will have no effect if the font already possesses that attribute. In other words, using {\b} on a bold font will **not** make the font "bolder".

Format	Code	Example	Result
Bold	{\btext}	{\bBold} text	Bold text

Italic	{\itext}	{\iltalic} text	Italic text
Subscript	{\dtext}	P{\dso}	$P_{so}$
Superscript	{\utext}	X{\u2} + Y{\u2}	$X^2 + Y^2$
Symbol	{\stext}	e{\u(-{\sa}t)}	e ^(-at)
Underline	{_ <i>text</i> }	{_underlined} text	underlined text
Enlarged	{\2text}	{\2{\sS}}x	Σx
Color	{\cBBGGRR <i>text</i> }	{\c0000FFRed} Text	Red Text
Overbar	{\otext}	{\oY}=56.23	<u></u> ¥=56.23
Dot accent	{\. <i>text</i> }	$Flow = \{ . V \}$	Flow = V
Double dot accent	<b>{\:</b> <i>text</i> <b>}</b>	$y = y{\d0} + {\.y}t - {\.y}t{\u2}/2$	$y = y_0 + \dot{y}t - \ddot{y}t^2/2$
Wingdings	<b>{\w1</b> <i>text</i> <b>}</b>	{\w1N} Danger {\w1N}	♣ Danger ♣
Wingdings 2	<b>{\w2</b> text <b>}</b>	{\w2R} Passed	⊠ Passed
Wingdings 3	{\w3text}	Arrows {\w3!"#\$} and\nmore arrows {\w3NOPQ}	Arrows ←→1↓ and more arrows ⊊⊃OO

# 💋 Related macro commands

DeleteRefLine RefLine

# 4.4.18 Generate Mesh command

Displays a dialog box that allows you to specify the size and number of intervals in a generated rectangular grid of 3D points based on the current 3D surface plot.

This routine uses one of two methods to generate Z values. **Quadratic** interpolation (in general) produces a smoother surface than what is typically displayed with randomly-spaced points. In many cases the resulting grid is a more accurate representation of the input than the triangular-mesh plot, but it should also be obvious to the user that no algorithm can make up for sparse data. **Quadratic** interpolation weights the result by the inverse of the square of the distance to the nearest **Control points** points. **Planar** interpolation simply maps each generated point to the planar surface of the triangle containing the point. The generated surface will not be as smooth as one produced by quadratic interpolation.

# Extrapolate points outside convex hull

Disabled for planar interpolation. If checked, DPlot will generate values for the entire extents (specified with the *from* and *to* boxes). Depending on the complexity of the existing surface near the edges, these extrapolated values may be, at best, questionable. If this box is **unchecked**, values outside the limits of the triangular mesh are set to -1.E+38, which in turn causes DPlot to ignore them.

# **Control points**

The number of points used to interpolate with the quadratic method. In most cases the default value of 4 works nicely. However in some cases more points are needed to smooth the resulting plot.

# 烤 Related macro commands

**GenerateMesh** 

# 4.4.19 Contour Options command

Displays a dialog box that allows you to specify the appearance of contour plots for 3D and 4D data. To edit the Z values of plots on a rectangular grid, or any of the X,Y,Z components on a plot of random points, use the Edit Data command on the Edit menu or click the Edit Data button on the toolbar:

Edit Data button:

Туре

■ Shaded bands- represents ranges of Z with different color bands.



□ Contour lines - draws contour lines at user-specified intervals (2D views only)







■ Both shades and lines (2D views only)

■ Waterfall plot - draws lines along the surface at constant values of X.



This option is only valid for gridded data (as opposed to randomly-spaced 3D points), and is disabled if "View in 3D" is unchecked. **NOTE:** To reorganize your randomly-spaced 3D points into a rectangular grid, use the <u>Generate Mesh</u> command.

■ Wireframe - draws grid lines and/or element edges ("Borders") along the surface. No hidden line/ surface removal.



This option is disabled if "View in 3D" is unchecked. Lighting, contour interval, and "Draw sides" settings are ignored. If both the "Grid" and "Borders" options under "Draw" are unchecked, border lines (lines delineating each grid rectangle or triangle) will be drawn. Line width is controlled by the "Wireframe lines" setting under "Other Lines".

Bar chart



The Waterfall and Bar Chart options are disabled for random 3D points; these options are only valid for Z values on a rectangular grid. To produce an interpolated rectangular grid from random 3D points use the <u>Generate Mesh</u> command on the Options menu.

### View in 3D

If checked, the plot is drawn with a 3D projection using the view angles specified by "Azimuth" and "Elevation". If unchecked, the plot is drawn as a plan view of the XY plane.

## Example:





z=cos(x)*sin(y)



**Azimuth** - view angle around the z axis. 0 degrees is looking in the positive X direction; 90 degrees is looking in the positive Y direction.

**Elevation** - view angle from the horizontal. 0 degrees gives an elevation view; 90 degrees is looking straight down at the surface. For a top-down view with X axis at the bottom of the plot and the Y axis labels along the left side of the plot, use elevation and azimuth both equal to 90 degrees.

Also see <u>Shortcuts</u> below for keyboard shortcuts to change the azimuth and/or elevation angles while viewing the plot.

Angles control - the graphic below the Elevation text box allows you to select viewing angles with

your mouse. Click within the circle to change the viewing angles. The center of the circle represents an elevation angle of 90 degrees (looking straight down at the surface), the inner circle is 45 degrees, and the outer circle is 0 degrees (viewing along the XY plane). The 3D unit axes will rotate as you move the mouse across this control. View angles will not be changed until you **click** within the circle.

You cannot select a negative elevation angle (looking up at the surface from underneath) using this control. To use a negative elevation angle enter the value in the **Elevation** text box or click on the corresponding spin button. (Note, though, that negative elevation angles are not recommended for most plot types.)

**Perspective** - if set to a positive value, determines (roughly) the distance from the viewpoint to the center of an imaginary box surrounding the surface for a perspective projection. This value is a multiplier for one-half the diagonal of that box. If less than or equal to 0, a normal orthographic projection will be drawn, with no consideration for perspective. DPlot will force a positive perspective value to a minimum of 1.0 (viewpoint on the surface of the smallest sphere that surrounds the plot). That minimum value typically distorts the plot to a larger degree than you will find desirable. Larger values will typically result in better-looking plots. A good starting point for the perspective value is 2.0.

Note: Perspective projection typically does not work well with high elevation angles and bar charts.

#### **Example:**

perspective=2



**Preserve Scale** - If checked, then the next time this plot is drawn the scale factor used to force the plot into the specified extents (or the available document area if extents are not specified with <u>Extents/Intervals/Size</u>) will be used on all subsequent renderings when spinning the plot around using the keyboard shortcuts <u>described below</u>. This option is primarily useful for smooth animations; if this option is **not** used then the plot size will tend to increase/decrease and possibly shift left, right, up, or down as you spin it around using the keyboard shortcuts. Note that this may mean that portions of the plot will be clipped when using this option. For best results first select viewing angles that will make the plot take up the largest portion of screen space - typically this will be at azimuths of 45, 135, 225, or 315 degrees.

- **Draw sides** If checked, planes will be drawn from the edges of the plot to the XY plane, obscuring the underside of the surface. You should **not** use an elevation angle less than 0 if **Draw sides** is checked. The edges of each plane will be drawn in the border color. Planes will be filled with the color specified by the control adjacent to the **Draw sides** checkbox.
- Example:



**Lighting** - choose from None, <u>Gouraud</u>, or <u>Phong</u> shading. Gouraud shading generally results in the surface having a faceted appearance. Phong shaded surfaces look much smoother, but require additional processor time and significantly more memory, particularly for very large data sets. Both models, as implemented in DPlot, assume a single light source in the direction of the viewpoint, infinitely far away. The effect of this assumption is that the light source vector is constant across the surface, and the light intensity is the same at all locations. No specular reflections are performed.

**□** Gouraud shading:





**Ambient light fraction** - values are limited to the range 0-1, inclusive. With 0 ambient light, planes that are near edge-on to the light source vector (the viewpoint) will be very dark. At the other extreme, a setting of 1 is equivalent to full brightness; the light source has no effect on appearance.

#### Lines/Levels

Intervals - Number of contour intervals to display. Limited to 255 bands or lines.

**Lower/Upper Limit** - Lowest and highest contour values. Keep in mind that the interval between legend values will be different for shaded bands than for contour lines. For shaded bands the interval is (upper limit-lower limit)/(number of intervals). For contour lines the interval is (upper limit-lower limit)/(number of intervals).

**Use actual displayed limits** - Check this box if you would like for DPlot to automatically adjust the lower and upper contour limits when you zoom in/out and/or force the plot extents by using the Extents/Intervals /Size command on the Options menu. If checked, the **Intervals** entry will be used as a *guideline*; the actual number used may be slightly more or less. If you zoom in to such an extent that only one data point is visible, or if all visible data points have the same magnitude, the lower and upper limits will be forced to that value +/- 5%, or +/- 0.5 in the case of Z=0. Leave this box unchecked to specify the lower and upper limits, which will then remain in effect on zoomed plots.

**Custom Levels** - If checked, then the **Intervals**, **Lower limit** and **Upper limit** settings are ignored and you may enter up to 255 unevenly-spaced contour values by clicking the adjacent >> button. This feature might be handy when a surface plot includes important details in specific Z ranges and those ranges are several orders of magnitude smaller than the Z extents.

**Results** - Shows the top five contour intervals that will be displayed, so you don't need to perform the division mentioned above in your head.

**Use > and/or < in legend** - If checked, for shaded bands if the maximum amplitude is more than 1 contour interval greater than the high contour value, DPlot adds  $a \ge prefix$  to the high value in the legend. It will use the same logic for the low value and  $a \le prefix$ . If unchecked, these prefixes will not be added. This setting only effects shaded bands, not contour lines.

#### **Contour lines**

Line width - Controls the width of the pen used to draw contour lines. **NOTE:** Colors and/or gray shades may not be printed correctly with very small line widths if your printer must simulate colors with dithered patterns. For example, an HP LaserJet uses patterns of black and white dots to simulate various colors and gray shades. Similarly, many color printers simulate a wide range of colors by using patterns of cyan, magenta, and yellow dots. If the pattern used by the printer to produce these simulations is wider than the specified line width, the output will not be as you expected. In particular, a 0.0" line width (1 printer dot) may result in a single white (invisible) line. For best results when printing colors or gray shades, the minimum line widths for all curves and grid lines should be set to no less than 12 dots, or 0.02 inches on a 600 dpi printer or 0.04

inches on a 300 dpi printer.

If you specify a line width less than 0 for the major or minor lines, the corresponding lines will not be drawn.

- Label every... If checked, DPlot will draw the contour values on a 2D contour line plot, regardless of whether **Display Legend** is checked. Control the frequency of the labels with the adjacent text box.
- **Transparent** If checked, the background of labels is not drawn. Otherwise a white background is used, usually obscuring the underlying contour line. (This option is ignored for "Both lines and shades", for which labels will **always** be opaque, preserving the background color.)
- Black text If checked, labels are drawn in black. Otherwise labels are drawn using the same color used for the corresponding contour line. (This option is ignored for "Both lines and shades", for which labels will always be drawn in black.)
- Horizontal If checked, all labels will be horizontal. If unchecked, labels will be oriented with the respective contour line at the point the label is drawn.
- **B/W based on luminance** This setting is only applicable when "Both lines and shades" is used. If the luminance (red*0.3 + green*0.59 + blue*0.11) of the region **above** the contour line is some threshold value or greater, the label is drawn in black. Otherwise the label is drawn in white.

#### Bar chart widths

These entries are disabled if *Bar chart* is not selected under *Type*. These values are the widths of bars as a fraction of the spacing between points in the X and Y directions. Valid entries range from 0.0-1.0.

#### **Example:**





# **Scale Factors**

Specifies how the X, Y, and Z extents are stretched or compressed relative to each other. These values are not related to the physical size of the plot. For example 1:2:4 results in the same effect as 3:6:12. This has no effect on the actual data values, only on appearance. You can have DPlot automatically initialize these scale factors to give square extents using the <u>General</u> command on the Options menu. Use a negative scale factor to invert the normal drawing direction.

#### Draw

**Axes** - If checked, the coordinate planes are drawn as a series of grid lines on 3D plots, and a box with X and Y scales is drawn around 2D plots. X and Y axis labels will not be drawn unless "Draw axes" is checked.

**Grid** - Draws grid lines across the plot at the same interval as the X and Y axis labels. The frequency of these tick marks can be changed with the Extents/Intervals /Size command on the Options menu.

**Borders** - Draws lines through the contour control points. For random 3D points this will draw the borders of the triangles. For points on a rectangular grid this results in lines around each grid cell. For bar charts, if this option is checked then bar edges are drawn in the border color. If unchecked bar edges are not drawn.

**Data points** - Draws filled black squares in the specified size at each data point. For all output devices, the square is rounded to the nearest odd number of pixels/printer dots, so that it will be centered on the point. Squares will be at least 3 pixels/dots wide and tall regardless of the size setting. Ignored for waterfall plots and bar charts.

**Legend** - Displays a legend using either color bars for "shaded bands" or colored text for "contour lines".

**Label every** <u>interval</u> - Controls the frequency of contour labels in the legend. If set to 1, every contour line will have a corresponding label in the legend. If set to 2, every other contour line will have a corresponding label, etc.

#### Colors

Choose between a smooth gradation of colors ranging from magenta to red, black to very light gray, or a custom blend of up to 64 colors that you specify.

**NOTE:** On 2D contour plots, if you use a custom color palette with more than one color and set the color for the low and high levels to the same color, then DPlot considers the contour values to cycle; in other words it considers the low end of the scale to be equivalent to the high end of the scale. In this case, triangles that have all three vertices at either the high or low end of the scale will use the same solid color across the entire face of the triangle. In the case of contour lines, no lines will be
drawn on this triangle.

### Related macro commands

Contour3D **ContourAxes ContourBarWidth** ContourColorScheme ContourCustomLevels **ContourDrawSides** ContourGrid **ContourLabelLines ContourLegend ContourLegendFrequency ContourLevels** ContourLighting **Contour**Method **ContourResetExtents ContourScales ContourView ContourViewChange SymbolSize SymbolType** 

#### Shortcuts (3D views):

CTRL+ ←	Decrease azimuth by 1 degree		
CTRL+ →	Increase azimuth by 1 degree		
CTRL+∔	Decrease elevation by 1 degree		
CTRL+†	Increase elevation by 1 degree		
CTRL+SHIFT+ ←	Decrease azimuth by 5 degrees		
CTRL+SHIFT+ →	Increase azimuth by 5 degrees		
CTRL+SHIFT+↓	Decrease elevation by 5 degrees		
CTRL+SHIFT+†	Increase elevation by 5 degrees		

**Please note:** These shortcuts work best with simple plots: no more than a few thousand data points, antialiasing and lighting turned off. Surface plots tend to be sluggish as DPlot draws each individual pixel. If looking for the best viewing angle, consider using the right-click Wireframe option, rotating the plot to the desired orientation, then switching back to a surface plot with that same right-click menu.

# 4.4.20 Scatter Plot Options

Displays a dialog box that allows you to specify the appearance of 3D and 4D scatter plots. To edit any of the X,Y,Z components on a plot of random points, use the <u>Edit Data</u> command on the Edit menu.

#### View

**Azimuth** - view angle around the z axis. 0 degrees is looking in the positive X direction; 90 degrees is looking in the positive Y direction.

**Elevation** - view angle from the horizontal. 0 degrees gives an elevation view; 90 degrees is looking straight down at the surface. For a pseudo-2D top-down view with X axis at the bottom of the plot and the Y axis labels along the left side of the plot, use elevation and azimuth both equal to 90 degrees.

Also see <u>Shortcuts</u> below for keyboard shortcuts to change the azimuth and/or elevation angles while viewing the plot.

Angles control - the graphic below the Elevation text box allows you to select viewing angles with

your mouse. Click within the circle to change the viewing angles. The center of the circle represents an elevation angle of 90 degrees (looking straight down at the surface), the inner circle is 45 degrees, and the outer circle is 0 degrees (viewing along the XY plane). The 3D unit axes will rotate as you move the mouse across this control. View angles will not be changed until you **click** within the circle.

You cannot select a negative elevation angle (looking up at the surface from underneath) using this control. To use a negative elevation angle enter the value in the **Elevation** text box or click on the corresponding spin button.

**Perspective** - if set to a positive value, determines (roughly) the distance from the viewpoint to the center of an imaginary box surrounding the surface for a perspective projection. This value is a multiplier for one-half the diagonal of that box. If less than or equal to 0, a normal orthographic projection will be drawn, with no consideration for perspective. DPlot will force a positive perspective value to a minimum of 1.0 (viewpoint on the surface of the smallest sphere that surrounds the plot). That minimum value typically distorts the plot to a larger degree than you will find desirable. Larger values will typically result in better-looking plots. A good starting point for the perspective value is 2.0.

**Preserve Scale** - If checked, then the next time this plot is drawn the scale factor used to force the plot into the specified extents (or the available document area if extents are not specified with <u>Extents/Intervals/Size</u>) will be used on all subsequent renderings when spinning the plot around using the keyboard shortcuts <u>described below</u>. This option is primarily useful for smooth animations; if this option is **not** used then the plot size will tend to increase/decrease and possibly shift left, right, up, or down as you spin it around using the keyboard shortcuts. Note that this may mean that portions of the plot will be clipped when using this option. For best results first select viewing angles that will make the plot take up the largest portion of screen space - typically this will be at azimuths of 45, 135, 225, or 315 degrees.

### **Scale Factors**

Specifies how the X, Y, and/or Z values are stretched or compressed. This has no effect on the actual input values, only the proportions of the plot. Use a negative scale to invert the normal drawing direction, e.g. use a negative Z scale to draw the plot with positive Z in the down direction.

### **Color Based On**

**Z Value** - Colors used to draw symbols are based on the Z value of the point. For this option, the number of levels and the high and low values are set in the **Z parameters** box.

Data Set - Each data set is drawn in the same color. You can have up to 100 data sets in a scatter plot.

View Depth - Colors are based on the distance of a symbol from the viewpoint.

#### Colors

Choose between a smooth gradation of colors ranging from magenta to red, black to very light gray, or a custom blend of 6 colors that you specify.

#### 📁 Related macro commands

ContourAxes ColorBasis ContourColorScheme ContourCustomLevels ContourLevels ContourScales ContourView ContourViewChange ScatterPlotLines

Shortcuts (3D views):

CTRL++ Decrease azimuth by 1 degree CTRL+ → Increase azimuth by 1 degree Decrease elevation by 1 degree CTRL++ CTRL++ Increase elevation by 1 degree CTRL+SHIFT+ ← Decrease azimuth by 5 degrees CTRL+SHIFT+ → Increase azimuth by 5 degrees CTRL+SHIFT++ Decrease elevation by 5 degrees CTRL+SHIFT++ Increase elevation by 5 degrees Please note: These shortcuts work best with simple plots: no more than a few thousand data points and antialiasing turned off.

# 4.4.21 Symbol Styles command (3D scatter plots)

### Symbol types

Select a symbol from the list, or click the "Other..." button to use a character from the Wingdings, Wingdings 2, or Wingdings 3 fonts. Note: These fonts must be installed on your system for this feature to work properly.

Note that there are two sets of inverted triangle symbols. The first two are placed such that the geometric center of the triangle is at the corresponding data point. The second two are placed such that they align vertically with the non-inverted (apex at the top) triangle symbols.

A cube symbol is available when the "Symbol size in Z units" box is checked. In that case, this symbol (unlike all others) will be drawn using the perspective settings for the plot. The perspective value may be set with the <u>Scatter Plot Options</u> menu command.

# Example:



# Symbol size

If the symbol size is set to 0.0 inches, DPlot uses a default symbol height of 1/40 of the height of the output device.

# Symbol sizes in Z units

If checked, the symbol size uses the same scaling as the Z values on the plot. If you zoom in/out or change the extents of the plot, the symbols will be resized accordingly. This option applies to **all** symbols on the plot; you cannot have symbols for one curve be a fixed physical size and symbols on another curve proportional to the plot amplitudes. Symbols in the legend will be drawn a fixed size, roughly the

same height as the accompanying text (if this option is unchecked then legend symbols will be the same size as those used in the plot).

#### Symbol frequency

Enter the frequency at which symbols are drawn. For example, if set to 1 then symbols will be drawn at every data point; if 2 then symbols are drawn at every other point. This option might be useful for plots with a large number of data points.

#### **Outline solid symbols**

If checked, solid filled symbols (not including Wingdings characters) will be outlined in the color selected below this checkbox. If unchecked, symbols are drawn with the color associated with that curve. The checkbox setting is applied to all curves; the color selection is specific to each curve.

# Shortcut:

Click 🗰 on the toolbar.

# 💋 Related macro commands

SymbolFillBackground SymbolFrequency SymbolOpacity SymbolOutline SymbolOutlineColor SymbolSize SymbolSizeUnits SymbolType

# 4.4.22 Box and Whisker Plot Format

Displays a dialog box that allows you to specify the plotting parameters for a box-and-whisker plot.

# Based on

Select from "Median-based" or "mean-based".

The center point in a median-based display is the group median, or the middle value. If the number of items in a group is odd, the median is the exact value of the middle number. If the number of items in a group is even, the median is the average of the two middle values. The median is also known as the value at the 50th percentile.

The center point in a mean-based display is the group average.

# **Median-based Options**

# **Box limits**

The limits of the box are typically the 25th and 75th percentiles. The distance between them is also known as the IQR, or interquartile range. The box limits value is the range between the ends of the box, centered around the 50th percentile.

### Whisker limits

The whiskers typically represent the "non-outlier" minimum and maximum values within the group, though you have several options for defining these limits. Whiskers will never be drawn beyond the limits of the data, regardless of the "percentage" entry .

# **Mean-based Options**

#### Limits function of:

Select from standard deviation or standard error (the standard deviation divided by the square root of the sample size). All plot limits are multipliers for this selection.

#### Limits expressed as % confidence

If limits are a function of standard error and this box is checked, then the limit input and legend display is expressed as a confidence level percentage, rather than a multiplier for the standard

error. This option is meaningless (and disabled) if the limits are instead a function of standard deviation.

#### **Box limits**

The limits of the box are typically the group standard deviation or the standard error.

# Whisker limits

The whiskers typically represent 2 or 3 standard deviations or standard errors.

#### Outliers

For median-based plots, outliers are typically defined as those individual points which are greater than 1.5 times the IQR, beyond the end of the box. Uncheck this box if you do not wish to display outliers.

For mean-based plots, outliers are typically defined as those individual points which are greater than 2 or 3 standard deviations from the mean. They are typically not plotted when the box or whiskers represent standard errors.

#### Extremes

For median-based plots, extremes are typically defined as those individual points which are 3 times the IQR, beyond the end of the box. Uncheck this box if you do not wish to display extremes.

#### Attributes

Click this button to display the <u>Box and Whisker Attributes</u> dialog, where you may specify attributes for the box, whiskers, mean/median, outliers and extremes.

# 4.4.22.1 Box and Whisker Plot Attributes

Displays a dialog box that allows you to specify the attributes (color, line/symbol styles, line widths, etc.) for a box-and-whisker plot.

#### Draw grand median/mean

Check this box to draw a solid line through the median (for median-based plots) or mean (for mean-based plots) for **all** groups.

#### **Vertical groups**

Check this box to rotate the plot such that groups are on the vertical axis and amplitudes are on the horizontal axis.

# Draw group info

Check this box to draw summary info (mean, standard deviation, and coefficient of variation) for each group.

# 4.4.23 Dot Graph Format

Displays a dialog box that allows you to specify the plotting parameters and attributes for a dot graph.

#### Group data points

Select from "Duplicates" and "Histogram".

#### Duplicates

Individual data points for each group are plotted along a single line (position determined by the value along the axis), and are stacked if there is an exact duplicate point.

#### Histogram

Individual data points for each group are plotted along a single line (position determined by the value along the axis), and are stacked according to which "bins" or "intervals" they fall in, as in a histogram. The formula for determining the bin or interval width is as follows:

interval width = 
$$\frac{R}{1+(3.322 \times \log n)}$$

where R is the range (maximum – minimum) of the entire plotted data set, n is the total sample size across all groups, and log is the natural logarithm.

The bins start with the lowest value in the entire data set, and continue up to the highest value in the data set. In terms of plotting the points, each individual data point will take on the value of the midpoint of the interval in which it falls.

### Draw grand median/mean

Check this box to draw a solid line through the median (for median-based plots) or mean (for mean-based plots) for **all** groups.

#### Vertical groups

Check this box to rotate the plot such that groups are on the vertical axis and amplitudes are on the horizontal axis.

#### Draw group info

Check this box to draw summary info (mean, standard deviation, and coefficient of variation) for each group.

# 4.4.24 Define Boundary command

This command allows you to specify the X and Y boundaries of a surface plot generated from random 3D points. This command can also be used to create holes in a surface.

#### Why is this command needed?

By default, DPlot will **always** generate a convex mesh of triangles from random 3D points. A simple list of points does not contain sufficient information to determine what the boundary of the surface should be, so DPlot makes this simplifying assumption that in many cases is appropriate. Of course there are many cases where this assumption is **not** appropriate. For concave edges you can delete unwanted triangles with the *Delete Triangles* command on the Edit menu, but that process may be time-consuming and, pertinent for program developers, requires interaction from the end user . And if the X or Y coordinates of a surface are edited in any way then a new **convex** mesh will be generated, destroying any work done with the *Delete Triangles* command. The *Define Boundary* command handles all of those shortcomings:

The border may often be defined by many fewer points than there are excess triangles.

The border is persistent. Changes to the data that result in a new triangulation will not generate triangles outside the border.

This procedure, unlike *Delete Triangles*, can be performed from another program with a call to DPlot_3DBorder in <u>dplotlib.dll</u>.

#### Operation

The border may start at any location and proceed in either a clockwise or counterclockwise direction around the surface. Border points do not necessarily have to be on the surface. X and Y components of each point may be separated by commas, spaces, or tabs, one point per line. As you enter each line you will see the border being updated on the plot (2D only), drawn as a reverse video line.

You do not need to close the boundary; DPlot will automatically connect the last point to the first.

Along with typing the values, if you are viewing the plot in 2D then you can use the mouse to select points by left-clicking the desired location on the plot. Points selected with the mouse will be added at the current line in the text box if the caret is in the leftmost position; otherwise these points will be added/ inserted on the line following the caret. For best results with the mouse you might prefer to check the *Snap to nearest data point* box. If that box is checked, then the X and Y values of the data point closest to the mouse position will be used, rather than the converted mouse coordinates. *Snap to nearest data point* has no effect on values you type in, nor will checking this box modify values that are already present, regardless of how they were entered.

After entering all border points and selecting *OK*, DPlot will delete any triangles whose average X and Y value is outside the border. The average value is used as a check simply because it is mathematically trivial to compute and it is guaranteed to be inside the triangle. More specifically, for every triangle in the plot DPlot extends a horizontal line to the right, past the rightmost extreme of the data. It counts intersections of this line segment with all border line segments. If the number of intersections is odd, the triangle is inside the border; if even, the triangle is outside.

**Note:** Data points found to be outside the border are **not** deleted; only triangles are deleted. So if you later use the *Define Boundary* command to remove the border, the original surface will be restored.

#### Holes

Along with defining the exterior edges of your surface you can use this command to specify holes in your surface. The trick is to ensure that the path along the border from the exterior of the surface is exactly the same, but in the reverse direction, as the path from the hole back out to the exterior edge. As an example, this plot:



depicts peak pressure measurements across the face of a wall with an opening in the center of the wall. No gages were located within the opening, so the plot is at best misleading. If you use the *Contour Options* command to turn on the triangle borders then the opening location becomes more apparent:



You can use *Define Boundary* to remove all of the triangles inside the opening. In the example shown, the points picked as the border are numbered 1-12, starting at the lower left corner and proceeding counterclockwise around the surface. Note that point 4 (at the top of the surface) is the same as point 11 and point 5 (at the top of the hole) is the same as point 10. In short, the path taken from the exterior edge to a hole can be anywhere on the surface as long as the exit path lies along those same points. For best results (to avoid roundoff errors) the entrance/exit path should be vertical or horizontal.



Finally, here is the result of using *Define Boundary* to remove triangles from the opening:



# 4.4.25 Background Image command

This command allows you to specify an image (BMP, GIF, JPG, PNG, or TIF) that will be drawn within the plot extents. The image will be stretched to fill the plot exents, regardless of its width/height relative to the plot width/height. To delete an existing background, delete the text in the Filename text box.

Background images are only applicable to 2D views of your data, and are not used for <u>polar plots</u> or triangle plots.

### Use full extents of image

If checked, the entire image will be used. If unchecked you may specify the top-down coordinates (0 is at the top of the image, in other words) of the image to display. The left, top, right, bottom text boxes will be disabled until/unless you select an image. Valid values are 0 to width-1 for the left and right; 0 to height-1 for the top and bottom. The left value must be less than the right, and the top value must be less than the bottom.

### Force plot extents to match aspect ratio of image

If checked then the plot extents will be adjusted such that the image has a 1:1 pixel ratio. For example if the image is 800 x 600 pixels and this box is checked, then the width and height of the plot will be set to a 4:3 ratio. If "Specify plot size" was previously selected and the current settings do not match the image aspect ratio, then the maximum dimension will be used for the larger of the width/ height. If "Specify plot size" has not been previously selected then the maximum dimension will be set to 8 inches. This adjustment takes place at the time this command is used and is not persistent: Subsequently changing the plot size with the Extents/Intervals/Size menu command or by dragging the corners of the plot may result in a distorted background image. (If you do this unintentionally, you can easily restore the plot aspect ratio by again selecting the Background Image command and immediately clicking OK w/o making any other changes.)

#### Draw in foreground

If unchecked then the image is drawn before any other items. If checked then the image is drawn **after** all other items.

### Transparency

Global transparency value applied to entire image. Valid range is 0 (invisible) to 255 (opaque). Default value = 255 (opaque).

#### Background Color

Color that will **not** be drawn. This setting is disabled for 32-bit images with an alpha channel.

NOTE: The image is **not** saved to a DPlot file. Instead, the filename of the image is saved within the file. The absolute path to the image is saved **unless** 1) the image resides in the same folder as the DPlot file, or 2) the image resides in My Documents\DPlot\Images. This presents a problem if copying a saved DPlot file to another system: if the path to the image does not exist on the target machine then the image will **not** be found. For the best results on multiple systems, copy the image file to the same folder where you will subsequently save the DPlot file, or to My Documents\DPlot\Images, **before** you specify that image. If you do that, the image location within the DPlot file will not include a path and the file will be expected to reside in one of those two locations. When reading a DPlot file, images saved to the DPlot file's folder take precedence over images saved to My Documents\DPlot\Images.

Example: GPS recording with a background image from Google Earth.



This feature can also of course be used to fill the background with a gradient. The example below uses a 1-pixel wide background image, with **Force plot extents to match aspect ratio of image** unchecked.



NOTE: If you select a new image with the Browse button and the width is 10 pixels or less and the height is 100 pixels or greater, or vice versa, DPlot will automatically uncheck the **Force plot extents to match aspect ratio of image** box, under the assumption that this image is simply a gradient. If you subsequently check that setting, your preference will be preserved.

#### 烤 Related macro commands

SetBackgroundImage

# 4.4.26 Insert Image command

This command allows you to insert up to 4 images (BMP, EMF, GIF, JPG, PNG, or TIF) that will be drawn on the plot. Enhanced metafiles **must** have an extension of ".emf" to work properly with this command. To delete an existing image, delete the text in the Filename text box or click the Delete button.

These images are only applicable to 2D views of your data.

#### Location

#### Distance from top left corner

If selected, then the anchor point of the image is measured from the top left corner of the plot, positive Y values down. Default units are those set with the *Default dimensions* setting of the <u>General</u> command on the Options menu, but may be overridden with the appropriate suffix (" or in=inches, pt=points, mm=millimeters, cm=centimeters).

If using this option you should also select *Specify plot size* with the <u>Extents/Intervals/Size</u> command on the Options menu. Otherwise the image may change location as the plot is resized or rendered to a different device or file.

**NOTE:** Images placed outside the plot extents may be clipped on the display, depending on the plot size set with <u>Extents/Intervals/Size</u>. If an image is clipped and this is not desirable, decrease the size of the plot with <u>Extents/Intervals/Size</u>. Images copied to the clipboard (both metafile and bitmap versions) and bitmap images saved to disk will generally be expanded as necessary to include all images.

#### Data space

If selected, then the anchor point is in the X and Y units of your plot. In this case if X or Y are outside the plot extents, the image will **not** be drawn.

#### Lock aspect ratio at 1:1

If checked then the image will be drawn at the aspect ratio of the image's width and height. As you change either the width or height entries, the height or width will also be changed to maintain the same aspect ratio.

### Draw in background

If checked then the image is drawn before any other items. If unchecked then the image is drawn **after** all other items.

# Transparency

Global transparency value applied to entire image. Valid range is 0 (invisible) to 255 (opaque). Default value = 255 (opaque). This setting is not used for metafiles.

**NOTE:** The image is **not** saved to a DPlot file. Instead, the filename of the image is saved within the file. The absolute path to the image is saved **unless** 1) the image resides in the same folder as the DPlot file, or 2) the image resides in My Documents\DPlot\Images. This presents a problem if copying a saved DPlot file to another system: if the path to the image does not exist on the target machine then the image will **not** be found. For the best results on multiple systems, copy the image file to the same folder where you will subsequently save the DPlot file, or to My Documents\DPlot\Images, **before** you specify that image. If you do that, the image location within the DPlot file will not include a path and the file will be expected to reside in one of those two locations. When reading a DPlot file, images saved to the DPlot file's folder take precedence over images saved to My Documents\DPlot\Images.

**NOTE:** Images with an alpha channel (e.g. 32-bit PNG) generally do not work well with this feature due to resizing. Areas along boundaries between transparent and opaque regions will often have strange artifacts.

# 💋 Related macro commands

InsertImage DeleteImage

# 4.4.27 General command

#### Prompt to save changed plots

Normally DPlot will prompt you before exiting to ask whether you want to save the current plot if it has been edited in any way. Uncheck this box if you find this feature to be annoying.

### Allow undo

Check this box to allow you to undo the last 100 operations (using the Edit>Undo command) or redo previous undo operations. If you frequently work with large data sets, this feature may cause DPlot to be a bit more sluggish than you'd like, as the entire data set will be saved to disk every time you edit the data values in any way.

#### Always start maximized

If this box is checked, DPlot will always start maximized (the window will fill the display). If unchecked, DPlot will start with the same position and size that were in effect the last time you closed the program.

#### Always auto-adjust scale factors on 3D plots

If this box is checked, DPlot will always initialize the X, Y, and Z scale factors (normally set on the <u>Contour Plot Options</u> dialog) such that the plot is square in the XY plane and the Z extents appear to be about 1/2 that of X and Y. This is a useful feature if you frequently plot data sets with values in different units.

#### Always force symbols on/lines off for triangle plots

<u>Triangle plots</u> are normally drawn with symbols at each data point, rather than line segments between points. If this box is checked then DPlot will automatically select a symbol style for all data sets (if one is not already selected) and set the line style for each data set to "None".

#### Antialias objects on display output

If this box is checked, symbols and line segments drawn on the display monitor will be antialiased, generally improving the appearance. This option is disabled if the display color depth setting is 256 colors (8 bits per pixel) or fewer.





This option will cause DPlot to take longer than normal to render a plot, particularly with 3D scatter plots and XY plots with many thousands of points or a thousand or more symbols. You may prefer to leave this option unchecked while editing a plot, and turn the option on before saving the plot to a bitmap. Alternatively, if you sometimes work with large data sets in XY plots but for all other graphs would like to keep antialiasing turned on, enter the maximum number of points for which you want to keep antialiasing in the **For XY plots**,... box. The default value of 5000 points generally works well.

#### Initially turn symbols off/lines on

If this box is checked, then for new XY plots DPlot will initially force the symbol type for all curves to none and the line type (if the line type is set to none) to solid lines. If this box is **unchecked**, settings for line and symbol styles are transferred from the first open document window or from DPlot's settings file for the first document window opened. (Line styles and symbol style settings for that first open document are retained even if the document currently has no plot.) This setting has no effect on the "Always force symbols on/lines off for triangle plots" setting.

This feature will prevent frustration for users who typically plot very large (several hundred thousand point) XY plots but also on occasion create plots with symbols, since symbols (especially antialiased symbols) take considerably longer to draw than line segments.

#### Recalculate extents of XY plots after hiding/unhiding curve(s)

If this box is checked, then the plot extents will be recalculated after hiding or making visible one or more curves using the *Hide all curves but...* or *Unhide all curves* commands on the View menu or one of the *Hide* commands on the legend right-click menu, ignoring any invisible curves. If this box is unchecked, then the extents are unchanged; invisible curves will be accounted for if/when the extents are recalculated.

This option is ignored if no curves are visible.

### Adjust extents after applying transform to data values

If this box is checked, then when the <u>Operate on X</u> or <u>Operate on Y</u> menu commands are used: 1) if the transformation is only a function of the respective axis, e.g. with **Operate on X** and X=X*50+6 but not with X=X+Y, and 2) the transformation is applied to all curves on the plot, then manually-set extents and tick mark intervals on that axis are transformed as well. This option helps prevent producing unexpected and/or undesirable results, e.g. without this option multiplying all Y values by 10,000 with manually set tick marks produces 10,000 times as many tick marks. If the above two conditions are not both true or if an error occurs when applying the transform, then manual extents and tick marks are turned off. If this box is unchecked then the extents and tick mark interval settings are left alone. In the case of 3D plots, if this box is checked then manual extents and tick marks are turned off, regardless of the transform equation.

### Show coordinates in popup when mouse hovers over data point

If checked, the X, Y, (and Z for 3D data) values of the point under the cursor are shown in a popup window. This feature works only in 2D views of your graph. If you find this feature annoying, uncheck this box.



### X values consisting of times are always sequential

When reading a column of time values (e.g. 7:30:00 AM) from a text file or the clipboard, the times are translated internally to numbers. Midnight=0, noon=12, etc. If this box is checked then time values are adjusted, if necessary, such that the time in row N is greater than or equal to the time in row N-1. If this box is unchecked, the X axis value will wrap when the time passes through midnight. If your data consists of sequential times you should check this box. If your data consists of measurements taken at random times, not recorded chronologically, you should uncheck this box.

#### Automatically read preference file based on filename

If checked, then when opening a data file for a new document DPlot will check for the existence of a preferences file (previously saved with <u>Save Preferences</u>) with the same filename and extension of ". set", first in the folder where the data file resides, then in the default preferences folder (% APPDATA%\DPlot\Preferences). If found, that preferences file will be processed. This might be a handy option if you have data files which frequently change, but you don't want to go through the same formatting steps each time you open that file. This option is not applicable to DPlot files, which by definition contain all of the formatting options that would normally be saved to a preferences file. When opening multiple files, this option is only applicable to the first file selected.

#### Assume input dates are of the form d/m/y

When reading dates from text boxes, the clipboard, or a text file, DPlot looks for 3 separate fields (day, month, and year) and makes these assumptions:

If a field is non-numeric (i.e. Jan, January, Feb, etc.) then this field represents the month. The year is assumed to be in the third field unless either of the first two fields is greater than 1000. If this box is unchecked then assume the date is in the order m/d/y unless the first field is greater than 12.

If this box is checked then assume the date is in the order d/m/y unless the second field is greater than 12.

#### Disable zooming with mouse wheel

If unchecked, you can zoom in/out on a 2D view of a plot with your mouse wheel. Some users may find this feature annoying, unintentionally zooming in on a plot. If this box is checked then this feature is disabled.

#### Click and drag to zoom

If checked, you can zoom in on a 2D plot by clicking and dragging your mouse. You can restrict

zooming to the X direction by first pressing the CTRL key, or to the Y direction by first pressing the SHIFT key.

If unchecked, clicking and dragging pans the plot. Whether this option is checked or not, you can pan a 2D plot with the keyboard arrow keys.

# Disable hide/unhide with middle mouse button

If unchecked, you can:

- hide a curve by clicking the middle mouse button while the mouse points to a curve.
- hide all curves **except** the one pointed to by pressing CTRL + the middle mouse button.
- make all curves visible by clicking the middle mouse button while the mouse does not point to a curve.

Check this option if you find that you frequently click the middle mouse button accidentally.

#### **Default dimensions**

This selection determines the default units used for entering line widths, symbol sizes, plot size, etc. You can override the default by supplying the appropriate unit suffix to numbers: " or in=inches, pt=points, mm=millimeters, cm=centimeters. Note that this entry has no effect on how these sizes are stored internally or saved to DPlot files, only on how these values are presented.

#### **Open/Save default view**

These options are available on Windows XP systems **only**. These settings determine the initial view type (thumbnails, icons, details, etc.) and sort order for the Windows common Open and Save dialogs shown by DPlot. For XP and later Windows versions, if you select a different view then that selection becomes the new default. However, the sort order will always revert to the selection shown on this dialog initially (because there is no good way to monitor and/or get that setting) for XP. The *Sort in descending order* setting is only meaningful when the view is set to *Details*. All other views will use ascending order. These options are not shown for Windows Vista and later versions because the OS (unlike XP) preserves the selected settings anyway, which makes these options somewhat redundant.

#### Max. number of header lines in multi-column text files

When first installed, DPlot will skip past up to 30 header lines in text files containing columns of data (file types D and K on the *Select File Type* dialog). Increase this value if your files contain more than 30 lines of text before the start of the data. This value should **not** be adjusted unless necessary: when you drag-and-drop files onto DPlot or open files in Explorer with **Open With**, DPlot must determine the file format. A larger value in this box than necessary means that DPlot will needlessly read that many lines from these files before determining that they are not multi-column text files.

#### **Decimal symbol**

When first started, DPlot will use the decimal symbol set with the Windows Control Panel. You can change that symbol here, independent of the Control Panel setting. If a comma is used, then multiple arguments to functions in  $\underline{Y=f(X)}$  and similar commands **must** be separated by semicolons. This setting has no effect on macros or on calls to <u>DPLOTLIB</u> functions from another program: The decimal symbol in that context must be a period '.', and arguments must be separated by commas. Text file output and saving data to the Clipboard is also not affected by this setting: DPlot will use a '.' for the decimal symbol and CSV files will use a comma as the column delimiter.

#### **Date separator**

When first started, DPlot will use the date separator from Control Panel, which is generally tied to a user's location and may be changed with regedit. You can change that symbol here, independent of the Control Panel setting. This setting might be important for German users (default date separator = '.') who work primarily with text files using a dot as the decimal separator (but no dates, or at least no dates using a '.' separator).

# 烤 Related macro commands

GeneralOptions SetDecimalSymbol

# 4.5 Generate Menu

This menu presents several options for generating new data based on one or more currently existing data sets. Each of these commands produces a new curve (preserving the original curves).

#### Average All Curves

Finds average value at each abscissa for two or more curves.

# Median of All Curves

Finds the median amplitude at each abscissa for two or more curves.

#### **Polynomial Curve Fit**

Least-squares curve fit.

#### Integrate a Curve

Find area under curve.

# **Differentiate a Curve**

Find slope of a curve.

#### Switch independent variable

Substitute the amplitudes from one curve for the independent variable (X values) for all other curves. Show results in a new document.

#### <u>Y=f(X)</u>

Generate a curve whose amplitudes are a function of X.

### X=f(T), Y=g(T)

Generate a curve from parametric equations with X and Y both functions of an independent parameter T.

#### <u>Z=f(X,Y)</u>

Generate a contour plot with elevation (Z) a function of X and Y.

#### X=f(T), Y=g(T), Z=h(T)

Generate a 3D scatter plot from parametric equations with X, Y, and Z all functions of an independent parameter T.

#### Y=Poly(Y)

Applies a polynomial transformation to the amplitudes of a curve.

# <u>Y=f(X,Y1,Y2,...)</u>

Generate a curve whose amplitudes are a function of X and/or one or more amplitudes from existing curves.

# <u>FFT</u>

Performs Fast Fourier Transform on a selected curve.

#### **Histogram**

Finds the frequency distribution for the amplitudes of a curve.

#### Average Curve with Error Bars

Produces a new plot with Y values equal to the mean at every X and error bars equal to the extremes, the standard deviation at any X, or the standard error at any X.

# **3D Surface Plots**

### Scatter Plot

Creates an equivalent 3D scatter plot from the existing 3D surface plot.

# XY Plot from 3D surface

Creates a new XY plot from the intersections of a specified vertical plane with the 3D surface.

#### Find volume under surface

Finds the volume between a 3D surface and a plane with user-specified constant Z value.

#### Find best-fit plane

Finds the equation of the plane that most closely matches the 3D surface.

#### Subtract a plane

Modifies the amplitudes of a 3D surface by subtracting a plane.

#### Z3=f(Z1,Z2)

Subtract, add, multiply or divide the amplitudes of two existing surfaces to produce a new surface.

#### Average surfaces

Generates a new 3D surface with amplitudes equal to the average amplitude for all selected surfaces.

# **3D Scatter Plots**

# Surface Plot

Creates an equivalent 3D surface plot from an existing 3D scatter plot, either a single data set or a combination of all data sets. Points with identical X,Y coordinates will be removed, preserving the point with the maximum Z value.

#### **Plugin modules**

Optional modules that extend the capabilities of DPlot.

See also: **Resize Arrays** 

#### 4.5.1 Average All Curves command

Produces a new curve with amplitudes equal to the average value of the amplitudes of the existing visible curves. The new curve will have the identical X values as the first visible curve. For all other curves, DPlot interpolates the amplitude at a given X. If a given X value is outside the extents of a curve, that curve is ignored for this particular X value.



AverageAll AverageAndReplace

#### 4.5.2 Median of All Curves command

Produces a new curve with amplitudes equal to the median value of the amplitudes of the existing visible curves. The new curve will have the identical X values as the first visible curve. DPlot interpolates the amplitudes of all other curves at a given X, sorts those amplitudes, and sets the value of the new curve to the median of those sorted values. If the number of amplitudes is even (2,4,6,etc.) DPlot uses the average of the two middle values.



**Medians** 

#### **Polynomial Curve Fit command** 4.5.3

#### Shortcut:

Click on the toolbar

Uses least-squares curve fitting procedures to find the polynomial of a user-specified order that best fits the data. A least-squares curve fit is one in which the sum of the square of the errors between the actual data and the polynomial model are minimized:

For N data points  $(x_i, y_i)$ , i = 1, ..., Nand a polynomial model for the data :  $y(x) = A_0 + A_1 x + A_2 x^2 + ... + A_m x^m$ , minimize:  $\sum_{i=1}^{N} [y_i - y(x)]^2$ 

DPlot converts the input curve to a form most likely to achieve a good fit (or at least most likely to avoid scaling errors), depending on the type of scaling currently in effect. The actual form of the output polynomial is dependent on the scaling of the plot:

Linear X, Linear Y:

 $y = C_0 + C_1 x + C_2 x^2 + \dots + C_m x^m$ 

Linear X, Logarithmic Y:

 $log_{10}y = C_0 + C_1x + C_2x^2 + \dots + C_mx^m$ or  $y = 10^{(C_0 + C_1x + C_2x^2 + \dots + C_mx^m)}$ 

Linear X, Probability Y:

NORMINV(y,0,1) = 
$$C_0 + C_1 x + C_2 x^2 + ... + C_m x^m$$

or

$$y = NORMDISTCDF((C_0 + C_1 x + C_2 x^2 + ... + C_m x^m), 0, 1)$$

Logarithmic X, Linear Y:

$$y = C_0 + C_1 \log_{10} x + C_2 (\log_{10} x)^2 + \dots + C_m (\log_{10} x)^m$$

Logarithmic X, Logarithmic Y:

$$\log_{10} y = C_0 + C_1 \log_{10} x + C_2 (\log_{10} x)^2 + \dots + C_m (\log_{10} x)^m$$

Logarithmic X, Probability Y:

NORMINV(y,0,1) =  $C_0 + C_1 \log_{10} x + C_2 (\log_{10} x)^2 + ... + C_m (\log_{10} x)^m$ 

Probability X, Linear Y:

$$y = C_0 + C_1 \text{NORMINV}(x,0,1) + C_2 (\text{NORMINV}(x,0,1))^2 + ... + C_m (\text{NORMINV}(x,0,1))^m$$

Probability X, Logarithmic Y:

$$\log_{10} y = C_0 + C_1 \text{NORMINV}(x,0,1) + C_2 (\text{NORMINV}(x,0,1))^2 + \dots + C_m (\text{NORMINV}(x,0,1))^m$$

Probability X, Probability Y:

NORMINV(y,0,1) = 
$$C_0 + C_1$$
NORMINV(x,0,1) +  $C_2$ (NORMINV(x,0,1))² + ... +  $C_m$ (NORMINV(x,0,1))^m where log₁₀ is the base 10 logarithm and NORMINV is the inverse of the cumulative distribution function.

Scale types not specifically mentioned above will use the default linear X, linear Y form.

Note that these conversions performed by DPlot will yield, for example, a straight line for a 1st order fit on any of the scale types mentioned above, though the relationship between X and Y may not be linear. If you **want** the standard polynomial  $y = C_0 + C_1x + C_2x^2 + ... + C_mx^m$  for a non-linear scale then you should switch to a linear X, linear Y scale before performing the curve fit, then back to the appropriate scale afterwards.

**Please note:** Points with Y values outside the limits set with <u>Amplitude Limits</u> are ignored. If you want those points to be considered, first turn Amplitude Limits off.

#### **Correlation coefficient**

The correlation coefficient presented by DPlot is a measure of how well the polynomial is correlated to the data. A correlation coefficient of 1.0 denotes a perfect fit, while a correlation coefficient less than 0.9 should probably be discarded. A high correlation coefficient less than 1.0 **does not necessarily** indicate an especially good curve fit (see below).

The correlation coefficient is described mathematically as:

$$Correl = \sqrt{1 - \frac{\sum_{i=1}^{N} [y_i - y(x)]^2}{\sum_{i=1}^{N} y_i^2 - \frac{1}{N} [\sum_{i=1}^{N} y_i]^2}}$$

### Standard error about the line

The standard error about the line is another measure of the goodness of the fit, and is analogous to standard deviation. A standard error of 0.0 indicates that the curve fit passes through the input data points exactly.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N} [y_i - y(x)]^2}{N - m}}$$

where m = the number of unknowns in the fit (1 for the mean, 2 for a straight line, 3 for a quadratic, etc.)

### Limitations

In general, least squares curve fits are very poor predictors of trends in the data outside the limits of the input. Extrapolation outside the limits of the input is not recommended for any but 1st order fits. The figure shown below illustrates the problem associated with extrapolating outside the limits of the input data. Even though the generated curve fits the input perfectly, values outside the limits of the data are, at best, suspect.



Bear in mind that the least squares procedure operates only on existing data points. A generated curve fit might fit the data points exactly, but produce completely unexpected results between data points. This is particularly true for high order polynomials. As in the figure above, the generated curve in the figure below fits the input data precisely. However, in this case, a 1st or 2nd order fit would produce a much more realistic model of the input data, even though the resulting polynomial would not be perfectly correlated.



A high (but less than 1.0) correlation coefficient is not necessarily an indication that a curve fit is appropriate for the input data. Consider the following examples, commonly referred to as <u>Anscombe's</u> Quartet:



The input data in each example has a mean Y value of 7.5, a mean X value of 9, a standard deviation in the Y values of 2.03, and a best fit line of Y=3+0.5*X with a correlation coefficient of approximately 0.81. It should be obvious from the above that the correlation coefficient alone is not a good predictor of the goodness of the fit or, especially, the goodness of the fit for one data set relative to another. You should **always** view a graphical representation of a curve fit before making any judgments concerning the appropriateness of that fit.

Related macro commands

**CurveFit** 

See also: Decimate

# 4.5.4 Integrate a Curve

DPlot finds the area under a curve using the trapezoidal rule:

 $y_{j,2} = y_{j-1,2} + (x_{j,1} - x_{j-1,1})(y_{j,1} + y_{j-1,1})/2$ 

Typical uses for integration include generating an impulse-time history from a pressure-time record, or velocity-time from acceleration-time.

For polar charts, DPlot uses

$$y_{j,2} = y_{j-1,2} + (x_{j,1} - x_{j-1,1})(y_{j,1} + y_{j-1,1})^2 / 8$$

(x in radians)

For a circle this will give an integrated value of (approximately, depending on the spacing between points)  $\pi r^2$  at x=360 degrees, as expected.



**Integrate** 

# 4.5.5 Differentiate a Curve

Three methods are available for differentiating (finding the derivative of) a curve. Typical uses for differentiating a curve include generating a velocity-time history from a displacement-time record, or pressure-time from impulse-time. The "Curve(s) to operate on" box contains a list of all curves with the minimum requirements: 2 or more points, and monotonically increasing X values. If only one curve exists or meets those requirements, it is selected and the "Curve(s) to operate on" list is disabled.

If your curves are not sorted in increasing X order and it is appropriate to do so, you may use the Sort command on the Edit menu so that those curves may be operated on by this command.

#### **Central Difference**

This method sets the output amplitude equal to the slope between the previous and next data points. For the first and last data points in the curve, the code uses the slope between the given point and the previous (for the last point) or next (for the first point) data point.

$$y'_{j} = \frac{y_{j+1} - y_{j-1}}{x_{j+1} - x_{j-1}}$$

This method gives results approaching the quadratic interpolation with small differences in spacing, and increasingly poor results for larger variations in spacing.

#### **Quadratic interpolation**

This method requires that the input data have a constant spacing. (If you select this option for a curve that does **not** have evenly-spaced points, then the **Central Difference** method will automatically be used.) DPlot uses a quadratic interpolating function that fits the amplitude at a given X plus its nearest two neighbors and sets the derivative at this X equal to the slope of the interpolating function.

#### **Previous point**

Very similar to "Central Difference", but uses the previous point and the current point (rather than the next point). The output for the first data point will use the first and second data points. Therefore the output amplitude at the first and second points will always be identical.

$$y'_{j} = \frac{y_{j} - y_{j-1}}{x_{j} - x_{j-1}}$$

This method might be useful for (among other things) rainfall intensity plots.

# **Replace existing curves**

If checked, the source curve will be replaced with the output of this function. If unchecked, this function creates a new curve for every source curve.

#### Legend

If the source curve has a <u>legend</u> entry, the output curve will have a legend of  $\frac{\partial}{\partial x}(<$ source legend>)".

# Related macro commands

**Differentiate** 

See also: Integrate a Curve

# 4.5.6 FFT command

# Shortcut:



Performs a Fast Fourier Transform (FFT) on a selected data record. The input data must be evenly spaced (have a constant interval).

If you believe that an interpolation of your input data is appropriate (it is outside the scope of DPlot to determine the answer to that question for you), you can use the <u>Equal Intervals</u> command on the Edit menu to generate a plot with a constant interval in X prior to using the FFT command.

Unlike the other Generate commands, the FFT command creates a new document window for displaying the output, since the input and output records are in different domains.

Given a function of time h(t) with N consecutive sampled values, the FFT implementation used by DPlot calculates the corresponding function in the frequency domain as:

$$H(f_{n}) = \frac{1}{N} \sum_{k=0}^{N-1} h(t_{k}) e^{2\pi i k n \cdot N}$$
  
where  
$$t_{k} = k\Delta$$
$$\Delta = time \ step$$

$$f_n = 0, K, \frac{1}{2\Delta}$$

DPlot can use either of two FFT algorithms, dependent on the number of points (N) in the input record:

# REALFT

Source: "Numerical Recipes" by W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, Cambridge University Press, 1986.

REALFT requires that the number of points (N) in the input record be a power of 2.

# EZFFTF

Developed by P.N. Swarztrauber, National Center for Atmospheric Research, Boulder, CO. Available on the World Wide Web at the NETLIB Repository, <u>http://netlib2.cs.utk.edu.</u>

EZFFTF does NOT have the same limitations on N, but can take significantly longer to compute the transform (see below).

The efficiency of the EZFFTF algorithm is highly dependent on the magnitude of the prime factors of N. For small prime factors, this routine is comparable in performance to the REALFT algorithm. However, for large prime factors, or for N equal to a large prime number, the run time of this routine can be several orders of magnitude longer than that for a comparable N consisting of small prime factors. The following table illustrates this.

N	Prime Factors	Run Time			
		90 MHz Windows 95	1.6GHz Windows 7		
16384	2^14	0.11 sec	< 1 msec		
16379	19*431	1.00 sec	187 msec		
16381	prime	97. sec	2.2 sec		
65536	2^16	0.50 sec	15 msec		
65521	prime	~33 minutes	34 sec		

If N is not a power of 2, you can elect to pad the input record with 0 values out to the next higher power of 2 by selecting the first option in "Adjustment Options" in the "FFT Options" dialog box. This operation will significantly decrease the run time for FFT's on input records with N equal to a large prime. **But**, this option may not always be appropriate, and it is not possible for DPlot to determine whether this feature is appropriate to use for a given data set or for a particular application. If in doubt and the prime factors are large, use "Run the FFT as is" and prepare to wait a bit.

If N is a power of 2, DPlot will always use the REALFT routine, regardless of the option selected.

If you are reasonably certain that you will always select the same options for performing an FFT, you can prevent the "FFT Options" dialog box from being displayed by selecting the "Don't show this dialog box..." check box. If you later decide that you need to change one or more options, exit DPlot, edit the file DPLOT.INI (in the Application Data\DPlot folder), and find the line "FFTOption=0xnnn", where nnnn is a hexadecimal number. Set this value to 0x0001 and start DPlot.

# X units

These entries are only important for figuring out the relationship between the X units and Hertz (for time units) or the X axis label of the generated FFT for "Other". DPlot guesses at the units on entry to this dialog, based on the X axis label for the input data. If the units are incorrect when you click OK, then the frequency label for the FFT will be incorrect. For example, if the input is in milliseconds but you select seconds, the frequency values will be kilohertz rather than Hertz.

# 烤 Related macro commands

<u>FFT</u> <u>FFTMethod</u> <u>FFTPhase</u>

# 4.5.7 Y=f(X) command

Generates X,Y pairs between a start and end point with a specified interval using a function that you specify.

Available operators are +, -, *, /, ^,>,<, and =. The inequality operators (>, <, and =) result in 1.0 if the condition is true, 0.0 otherwise.

Function (sin, cos, etc.) arguments must be in parentheses. See the list of available functions.

You can use placeholder codes for certain data parameters. For example "\$YMAX(2)" will be replaced with the maximum Y value of the second curve. For more information on placeholder codes click <u>here</u>.

You can create conditional equations with the **if** function. The **if** function takes the form if(*test*, *if_true*, *if_false*), where *test* is the condition you want to check, *if_true* is the value the equation takes if *test* is true, and *if_false* is the value taken if *test* is false. For example, "if x > 5 then  $y=x^2$ , otherwise  $y=(10-x)^2$ " can be expressed with **if**( $x<5,x^2,(10-x)^2$ ).

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute the value of pi (3.141592653589793...) for "PI", "pi", or "Pi"

NOTE: Be specific. DPlot does not interpret "2X" as "2*X".

Example: With **radians** selected, Y=sin(Pi*x) from X=0 to 2 results in:



#### Prevent divide by zero errors

If this box is **unchecked**, DPlot will trap divide by zero and many other math errors (e.g. logarithm of a non-positive value), report the value of x where the error occurred, and will not generate a curve. If this box is **checked**, when DPlot traps a math error it will shift x by a very small number and attempt the calculation again. This feature might be useful for generating curves of certain functions that would normally fail at specific values of x, e.g. sin(x)/x at x=0.

#### Constants

You can specify up to 10 constants to use in place of numerical values in your equation. Often the use of constants will make your equation more readable and less error-prone. For example, the Planck law for blackbody radiation as a function of wavelength is given by:

$$\mathsf{B}_{\lambda}(\mathsf{T}) = \frac{2\mathsf{hc}^2}{\lambda^5} \frac{1}{\mathsf{e}^{\mathsf{hc}/(\lambda\mathsf{KT})} - 1}$$

where

h =Planck's constant, 6.6262E-27 erg-sec

c = the speed of light, 2.99792458E+8 meters/sec

- k = Boltzmann's Constant, 1.3807E-16 erg/deg. K
- T = temperature, deg. K

Without using constants this equation can be expressed in DPlot as

 $y = 2*6.6262E-27*2.99792458E+8^2/x^5/(exp(6.6262E-27*2.99792458E8/(x*1.3807E-16*200))-1)$ 

(for T=200K) which is a bit hard to decipher. Instead, you can use

 $y = 2^{h*c^2/x^5/(exp(h^*c/(x^*k^*200))-1)}$  with c, h, and k defined as shown below:

Y = f()	()				? 🛛
Enter f(x) (click "Help" for example Y= 2*h*c^2/x^5/(exp(h*c/(x*k* Angles: radians degrees with interv ✓ Prevent divide by Constructs			ile us: (*200 (from) (to) rval <u>d</u> (0 en	age) ))-1) X= 1e-6 X= 5e-5 X= 1e-8 rors	Cancel <u>H</u> elp
Con	istants:				
A:	0		G:	0	
B:	0		H:	6.6262e-27	
C:	299792458		l:	0	
D:	0		J:	0	
F:	0		K:	1.3807e-16	

Repeating for several temperatures gives:



#### Expand constants in legend

If unchecked, the legend for a generated curve will generally match the equation as entered (plus a few formatting changes for subscripts and superscripts). If checked, any constants in the equation will be replaced by their numeric values. For example, with  $Y=A+B^*x+C^*x^2+D^*x^3+F^*x^4$  and *Expand constants in legend* unchecked, the legend will be  $A+B^*x+C^*x^2+D^*x^3+F^*x^4$ . With *Expand constants in legend* checked, the legend might be 1.111-0.4366*x+0.04882*x²-0.001627*x³+5.613e-016*x⁴. If this expansion results in a character string longer than 80 characters, the constants A,B, C,etc. will be used.

Tip: You can hide the legend by right-clicking and selecting *Hide Legend*. Move the legend by

clicking and dragging. Other legend parameters are accessible via the right-click menu, or by selecting the Legend/Labels command on the *Text* menu.

#### **Open/Save as**

These buttons allow you to read or save the entire setup for this function, including the equation, extents, constants, and other options. This might be handy if you have an operation that you frequently use **and** you use this command often enough that a particular operation migrates out of the most-recently-used list. We **strongly** suggest that you use the default folder location when saving a setup, so that you don't confuse the setup for this function with that for the X=f(T), Y=g(T) or Y=f(X,Y1,Y2,...) commands. (The saved files for the various Generate commands are not compatible.)

# 4.5.8 X=f(T), Y=g(T) command

Generates X,Y pairs from parametrical equations in which X and Y are both functions of a third independent parameter T. This command is useful for plotting curves in which Y cannot be expressed as a unique function of X. For example a circle may be plotted with X=cos(T), Y=sin(T). You might also find this command useful for generating unevenly-spaced data points. For example if you want your points to be evenly spaced on a logarithmic scale, you can use  $X=10^{-1}$  (T), Y=some function of ( $10^{-1}$ (T)).

Available operators are +, -, *, /, ^,>,<, and =. The inequality operators (>, <, and =) result in 1.0 if the condition is true, 0.0 otherwise.

Function (sin, cos, etc.) arguments must be in parentheses. See the list of available functions.

You can use placeholder codes for certain data parameters. For example "\$YMAX(2)" will be replaced with the maximum Y value of the second curve. For more information on placeholder codes click <u>here</u>.

You can create conditional equations with the **if** function. The **if** function takes the form if(*test,if_true*, *if_false*), where *test* is the condition you want to check, *if_true* is the value the equation takes if *test* is true, and *if_false* is the value taken if *test* is false. For example, "if t > 5 then  $y=t^2$ , otherwise  $y=(10-t)^2$ " can be expressed with **if**( $t<5,t^2,(10-t)^2$ ).

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute the value of pi (3.141592653589793...) for "PI", "pi", or "Pi"

**NOTE: Be specific.** DPlot does **not** interpret "2T" as "2*T".

Example: With **radians** selected, X=sin(2*T) Y=sin(3*T) from T=0 to 6.28 results in:



#### Prevent divide by zero errors

If this box is **unchecked**, DPlot will trap divide by zero and many other math errors (e.g. logarithm of a non-positive value), report the value of T where the error occurred, and will not generate a curve. If this box is **checked**, when DPlot traps a math error it will shift T by a very small number and attempt the calculation again. This feature might be useful for generating curves of certain functions that would normally fail at specific values of T.

#### Constants

You can specify up to 10 constants to use in place of numerical values in your equation. Often the use of constants will make your equation more readable and less error-prone.

#### Expand constants in legend

If unchecked, the legend for a generated curve will generally match the equations as entered (plus a few formatting changes for subscripts and superscripts). If checked, any constants in the equation will be replaced by their numeric values. If this expansion results in a character string longer than 80 characters, the constants A,B,C,etc. will be used.

**Tip:** You can hide the legend by right-clicking and selecting *Hide Legend*. Move the legend by clicking and dragging. Other legend parameters are accessible via the right-click menu, or by selecting the Legend/Labels command on the *Text* menu.

#### **Open/Save as**

These buttons allow you to read or save the entire setup for this function, including the equations, extents, constants, and other options. This might be handy if you have an operation that you frequently use. We **strongly** suggest that you use the default folder location when saving a setup, so that you don't confuse the setup for this function with that for the  $\underline{Y=f(X)}$  or  $\underline{Y=f(X,Y1,Y2,...)}$  commands. (The saved files for the various Generate commands are not compatible.)

# 4.5.9 Z=f(X,Y) command

Generates X,Y,Z triplets on a rectangular grid. If the active document contains a 3D scatter plot, a new data set will be created by this command and added to this document. For all other plot types, this command automatically opens a new window (if the current window contains a plot) and generates a contour plot of the generated data.

Function (sin, cos, etc.) arguments must be in parentheses. See the list of available functions.

You can use placeholder codes for certain data parameters. For example "\$ZMAX" will be replaced with the maximum Z value of the surface. For more information on placeholder codes click <u>here</u>.

You can create conditional equations with the **if** function. The **if** function takes the form if(*test*, *if_true*, *if_false*), where *test* is the condition you want to check, *if_true* is the value the equation takes if *test* is true, and *if_false* is the value taken if *test* is false. For example, "if x > 5 then  $z=x^2$ , otherwise  $z=(10-x)^2$ " can be expressed with **if**( $x<5,x^2,(10-x)^2$ ).

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute the value of pi (3.141592653589793...) for "PI", "pi", or "Pi"

**NOTE: Be specific.** DPlot does **not** interpret "2X" as "2*X".

#### Example:

```
With radians selected, Z=cos(X)*sin(Y)
from X=-3.14 to 3.14 and Y=-3.14 to 3.14 results in:
```



# Prevent divide by zero errors

If this box is **unchecked**, DPlot will trap divide by zero and many other math errors (e.g. logarithm of a non-positive value), report the value of X and Y where the error occurred, and will not generate a surface. If this box is **checked**, when DPlot traps a math error it will shift X and/or Y by a very small number and attempt the calculation again. This feature might be useful for generating curves of certain functions that would normally fail at specific values of X and Y.

**Note:** As far as DPlot is concerned, the grid produced by this operation retains the same constant spacing in X and Y, though X and/or Y may be shifted by a small value at one or more grid intersections.

# Overwite existing surface

If checked, the existing surface is deleted and replaced with the surface produced by this command, in the same document window. If unchecked, a new document is created. This box is hidden if the currently active document has no plot or if the plot for the active document is anything other than 3D points on a rectangular grid. For existing 3D scatter plots, the new data will be added to the existing graph as a new data set. For all other data types a new document will be created.

#### Constants

You can specify up to 10 constants to use in place of numerical values in your equation. Often the use of constants will make your equation more readable and less error-prone.

#### Expand constants in legend/title

If unchecked, the title for a generated surface (or the legend entry for a 3D scatter plot) will generally match the equation as entered (plus a few formatting changes for subscripts and superscripts). If checked, any constants in the equation will be replaced by their numeric values. (For surface plots: You can hide the legend by right-clicking and selecting *Hide Legend*. Move the legend by clicking and dragging. Other legend parameters are accessible via the right-click menu, or by selecting the <u>Legend/</u> <u>Labels command</u> on the *Text* menu.)

# 4.5.10 X=f(T), Y=g(T), Z=h(T) command

Generates X,Y,Z triplets that are all functions of an independent parameter T. If the active document contains a 3D scatter plot, a new data set will be created by this command and added to this document. For all other plot types, this command automatically opens a new window (if the current window contains a plot) and generates a 3D scatter plot of the generated data.

Function (sin, cos, etc.) arguments must be in parentheses. See the list of available functions.

You can use placeholder codes for certain data parameters. For example "\$ZMAX" will be replaced with the maximum Z value of the surface. For more information on placeholder codes click <u>here</u>.

You can create conditional equations with the **if** function. The **if** function takes the form if(*test,if_true*, *if_false*), where *test* is the condition you want to check, *if_true* is the value the equation takes if *test* is true, and *if_false* is the value taken if *test* is false. For example, "if T > 5 then  $Z=T^2$ , otherwise  $Z=(10-T)^2$ " can be expressed with **if**( $T<5,T^2,(10-T)^2$ ).

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute the value of pi (3.141592653589793...) for "PI", "pi", or "Pi"

NOTE: Be specific. DPlot does not interpret "2T" as "2*T".

Example: With radians selected, X=sin(2*T) Y=sin(3*T) Z=T from T=0 to 6.28 results in:



#### Prevent divide by zero errors

If this box is **unchecked**, DPlot will trap divide by zero and many other math errors (e.g. logarithm of a non-positive value), report the value of X and Y where the error occurred, and will not generate a surface. If this box is **checked**, when DPlot traps a math error it will shift T by a very small number and attempt the calculation again. This feature might be useful for generating curves of certain functions that would normally fail at specific values of T.

### Constants

You can specify up to 10 constants to use in place of numerical values in your equation. Often the use of constants will make your equation more readable and less error-prone.

#### Expand constants in legend/title

If unchecked, the legend entry will generally match the equation as entered (plus a few formatting changes for subscripts and superscripts). If checked, any constants in the equation will be replaced by their numeric values. You can hide the legend by right-clicking and selecting *Hide Legend*. Move the legend by clicking and dragging. Other legend parameters are accessible via the right-click menu, or by selecting the Legend/Labels command on the *Text* menu.)

# 4.5.11 Y=Poly(Y) command

This command applies a polynomial transformation to the amplitudes of a curve. The new curve has the form:

 $Y_{new} = C_0 + C_1 Y_{old} + C_2 Y_{old}^2 + \ldots + C_n Y_{old}^n$ 

# 4.5.12 Y=f(X,Y1,Y2,...) command

Generates X,Y pairs using a function of X and/or one or more Y values from existing curves. Y values from existing curves are included in the equation as "Yn", where n is the index of the curve. You are limited to the first nine existing curves (Y1, Y2, ... Y9). Curves used in the equation must have

monotonically increasing X values.

#### Use X values from curve

If checked, then the output curve will use X values from the 1-based curve index specified in the adjacent text box. If unchecked, the output curve will have evenly-spaced X values between the **from** X and **to X** values **with interval DX**.

If **Use X values from curve** is unchecked and you leave either the **from X** or **to X** boxes blank, DPlot assigns default values as follows:

If the equation includes Y terms, the **from X** value is set to the maximum starting X value for all curves included in the equation. The **to X** value is set to the minimum ending X value for those same curves.

Otherwise, if the document contains any curve, from X is set to the minimum starting X for all curves; to X is set to the maximum ending X value.

If the document contains no curves, from X is set to 0 and to X is set to 1.

For every generated X value, if the equation contains Y terms then DPlot interpolates the appropriate curve to find the value of Y to substitute in the equation. If X is outside the extents of the curve, then the corresponding Y value is set to 0.

# Prevent divide by zero errors

If this box is **unchecked**, DPlot will trap divide by zero and many other math errors (e.g. logarithm of a non-positive value), report the value of x where the error occurred, and will not generate a curve. If this box is **checked** and the equation has one or more X terms, when DPlot traps a math error it will shift x by a very small number and attempt the calculation again. This feature might be useful for generating curves of certain functions that would normally fail at specific values of x, e.g. sin(x)/x at x=0.

# Expand constants in legend

If unchecked, the legend for a generated curve will generally match the equation as entered (plus a few formatting changes for subscripts and superscripts). If checked, any constants in the equation will be replaced by their numeric values. For example, with  $Y=A+B^*x+C^*x'^2+D^*x'^3+F^*x'^4$  and *Expand constants in legend* unchecked, the legend will be  $A+B^*x+C^*x^2+D^*x^3+F^*x^4$ . With *Expand constants in legend* checked, the legend might be 1.111-0.4366*x+0.04882*x^2-0.001627*x^3+5.613e-016*x^4. If this expansion results in a character string longer than 80 characters, the constants A,B, C,etc. will be used.

# **Open/Save as**

These buttons allow you to read or save the entire setup for this function, including the equation, extents, constants, and other options. This might be handy if you have an operation that you frequently use **and** you use this command often enough that a particular operation migrates out of the most-recently-used list. We **strongly** suggest that you use the default folder location when saving a setup, so that you don't confuse the setup for this function with that for the  $\underline{Y=f(X)}$  or  $\underline{X=f}(T)$ ,  $\underline{Y=g(T)}$  commands. (The saved files for the various Generate commands are not compatible.)

Function (sin, cos, etc.) arguments must be in parentheses. See the list of available functions.

You can use placeholder codes for certain data parameters. For example "\$YMAX(2)" will be replaced with the maximum Y value of the second curve. For more information on placeholder codes click <u>here</u>.

You can create conditional equations with the **if** function. The **if** function takes the form if(*test*, *if_true*, *if_false*), where *test* is the condition you want to check, *if_true* is the value the equation takes if *test* is true, and *if_false* is the value taken if *test* is false. For example, "if x > 5 then  $z=x^2$ , otherwise  $z=(10-x)^2$ " can be expressed with **if**( $x<5,x^2,(10-x)^2$ ).

By default, angular input values are in radians. You can set angles to degrees by clicking the **degrees** option button.

DPlot will automatically substitute the value of pi (3.141592653589793...) for "PI", "pi", or "Pi"

NOTE: Be specific. DPlot does not interpret "2X" as "2*X".

#### Example:

This example shows a curve fit to input data and the square of the errors between the curve fit and the input. The third curve was generated with the equation "(Y2-Y1)^2".



**Tip:** You can hide the legend by right-clicking and selecting *Hide Legend*. Move the legend by clicking and dragging. Other legend parameters are accessible via the right-click menu, or by selecting the <u>Legend/Labels command</u> on the *Text* menu.

# 烤 Related macro commands

AddCurves Compare DivideCurves FunctionOfXY1Y2 FXY1Y2_XFrom Lump MultiplyCurves SubtractCurves

# 4.5.13 Histogram command

Use the Histogram command to plot the frequency distribution of the amplitudes of a selected record. The Histogram command creates a new document window.

DPlot initially suggests a sampling interval such that:

 $k = 1 + 3.3 \log_{10} n$ 

#### where

k = number of sampling intervals

n = total number of samples

This is the sampling interval suggested by H.A. Sturges in "The choice of a class interval," *Journal of American Statisticians Association*, vol. 21, 65-66, 1926

# Example

Interval between eruptions of Old Faithful measured in 2006. *Source: <u>http://www.geyserstudy.org/geyser.aspx?pGeyserNo=OLDFAITHFUL</u>* 



# Old Faithful Geyser Eruptions, 2006 Total of 5220 eruptions

The Histogram command with an interval of 5 minutes (expressed as either 0:05 or 0.083333 hours) yields:



The mean, standard deviation, and skewness properties are calculated automatically by the Histogram command. Skewness is a measure of the tendency of deviations from the mean to be larger in one direction:

skewness = 
$$\frac{\sum (X - \mu)^3}{N * \sigma^3}$$

where

 $\mu = mean$ 

 $\sigma$  = standard deviation

Right-click on the plot and select <u>Bar Chart Options</u> to change various aspects of the plot. For example the following plot was created by checking the "Percentage labels", "Labels outside bars", and "Shaded bars" boxes. The extents of the graph were changed with the <u>Extents/Intervals/Size</u> command on the Options menu.



As shown in this graph, 74.9% of Old Faithful's eruptions occurred in intervals between 1:25 and 1:40, and 91% of the eruptions occurred in intervals between 1:20 and 1:45 - presumably a reason for the geyser's name.

# Related macro commands

**Histogram** 

# 4.5.14 Average Curve with Error Bars

This command generates a new document showing mean (average) values of Y for all points within a user-specified interval in X. The command also creates error bars with a choice for the error bar amplitudes of:

- 1. Mean minus the extreme high and low amplitudes.
- 2. Standard error from the mean. For purposes of this command, standard error is the standard deviation divided by the square root of the number of points within any interval.
- 3. Standard deviation. Standard deviation is defined by

$$stdev = \sqrt{\frac{\sum (y_i - \overline{y})^2}{N - 1}}$$

where

 $y_i = amplitude of the i'th point$ 

$$\overline{y} = mean$$

N = number of data points

If the "Group X values within interval" value is set to 0, points are grouped only if the X coordinate is **exactly** the same. Otherwise input X values are rounded to the nearest interval.

The "Start at X=" entry determines the starting value of X in the output, with all subsequent X values being equal to that start value + some whole number times the interval. "Start at X=" is not used if the interval is set to 0 (no grouping).

Example:

With "High and Low extremes" selected and an interval of 2.0, "Start at X" = 0, this graph:


produces this result:



## 4.5.15 XY Plot from 3D Surface command

Creates a new XY plot from the intersections of a specified vertical plane with the 3D surface. The plane may be defined with

- The equation Y=mX+b. In a 2D view, a horizontal (constant Y) line has 0 slope (m=0) and b = the amplitude (constant Y).
- Constant X; i.e. a vertical line when viewed in 2D.
- A line passing through 2 x,y pairs. Alternatively to using the text boxes, with this option selected you can also click on the plot to select each point. These points are **not** necessarily the endpoints of the generated curve, they are only used to define the intersection plane.

After clicking OK, DPlot finds the intersections of the specified vertical plane with each triangle on the 3D surface plot. (To view these triangles, check "Borders" on the Contour Options dialog). For all but "X=constant", the dependent axis will be X values from your surface plot. For "X=constant" the dependent axis will be Y values from your surface plot. For non-zero slopes, the generated XY plot will include a second X axis with Y values from the intersection plane.

The generated XY plot will copy the first 2 title lines from your surface plot (if they exist) and the X and/ or Y and Z axis labels from your surface plot, or use default axis labels if the surface plot does not have axis labels. The 3rd title line will specify the intersection plane.

# 4.5.16 Find Volume Under Surface command

Finds the volume under a 3D surface above and below a reference Z value. If the reference Z value is above the surface, the calculated "above" volume is negative.

### Example:

For this surface:



Volume under 3D Surface 🛛 🔀	Volume under 3D Surface				
Finds the volume between the 3D Close surface and a plane with the specified Close	Finds the volume between the 3D Close constant Z.				
☑ Consider entire surface	☑ Consider entire surface				
Xlow: -3.14159 Xhigh: 3.14159	×low: -3.14159 ×high: 3.14159				
Ylow: -3.14159 Yhigh: 3.14159	Ylow: -3.14159 Yhigh: 3.14159				
NOTE: Only triangles completely within these bounds will be considered.	NOTE: Only triangles completely within these bounds will be considered.				
Base <u>Z</u> : 0 <u>Calculate</u>	Base Z: -1				
Volume <u>a</u> bove Z = 7.96712	Volume <u>a</u> bove Z = 39.4784				
Volume <u>b</u> elow Z = 7.96712	Volume <u>b</u> elow Z = 0				
<u>I</u> otal (above+below) = 15.9342	<u>I</u> otal (above+below) = 39.4784				
Difference (above-below) = 0	Difference (above-below) = 39.4784				

By default this operation will consider the entire surface. If "Consider entire surface" is unchecked, you may specify the X and Y extents within which the volume will be calculated. This might be useful for surfaces containing multiple peaks, where you want the volume under only one of those peaks. **Please note:** Triangles with one or more vertices outside those extents will be ignored.

## 4.5.17 Find Best-Fit Plane command

Uses least-squares procedures to find the plane that best fits the data. A least-squares fit is one in which the sum of the square of the errors between the actual data and the plane equation are minimized:

For N data points  $(x_1, y_1, z_1)$ , i = 1, ..., Nand a plane equation z(x, y) = Ax + By + C,  $\therefore \ \ldots \ \sum_{i=1}^{N} r_{-i} = -c_{-i} \dots N^2$ 

minimize 
$$\sum_{i=1}^{\infty} [z_i - z(x, y)]$$

### **Correlation coefficient**

The correlation coefficient presented by DPlot is a measure of how well the plane fits the data. A correlation coefficient of 1.0 denotes a perfect fit. The correlation coefficient is described mathematically as:

Correl = 
$$\sqrt{1 - \frac{\sum_{i=1}^{N} [z_i - z(x)]^2}{\sum_{i=1}^{N} z_i^2 - \frac{1}{N} \left[\sum_{i=1}^{N} z_i\right]^2}}$$

### 4.5.18 Subtract a Plane command

Subtracts a plane (z=Ax+By+C) from the amplitudes (z values) of a surface plot. You can specify the plane to subtract in one of three ways:

### Click on and/or type 3 points

Enter x, y, and z values for 3 data points or alternatively click on the plot (2D views only) to select one point after another. In the latter case DPlot will use the values of the data point closest to the mouse pointer. The three points must not be collinear. Once 3 points that uniquely define a plane are entered or selected, the equation of the corresponding plane will be shown below the text boxes.

### Specify plane equation

Enter the coefficients in the equation z=Ax+By+C.

### Use best-fit plane

Subtract the least squares plane that best fits the data. This is the same plane solved for by the *Find* best-fit plane menu command.

## 4.5.19 Z3=f(Z1,Z2) command

This command generates a new 3D surface with amplitudes equal to the result of operating on the amplitudes of two existing surfaces. That operation might be subtraction, addition, multiplication, division, finding the arctangent of surface A divided by surface B, or finding the minimum or maximum amplitudes of those surfaces.

The command is disabled if there are not two or more 3D surface plots open.

Given two surfaces A and B, DPlot produces a new plot with the X and Y coordinates from surface A. Z values from surface B are obtained from the plane equation of the triangle bounding those X, Y values. If a given point is outside the extents of surface B (in other words if there is no triangle that bounds that point), then the point is discarded from the generated surface.





This procedure will generate a new 3D surface that is the result of an operation on the amplitudes of two existing surfaces. Data in the generated surface will be at the same X and Y locations as surface A.       OK         If the xy coordinates of a point on surface A are outside the extents of surface B, OR if the operation would result in a math error (division by 0, for example) the new surface will not include that point.       OH         Surface A:       Surface B:         Cosx grf       Operation:
Surface A: Surface B: Operation: Cosx.grf
Cosx.grf Cosx.grf
Siny.grt OA-B
() A + B
A/B [↓] δ
C ATAN2(A,B)
MIN(A,B)
MAX(A.B)



## 4.5.20 Average Surfaces command

This command generates a new 3D surface with amplitudes equal to the average amplitude for all selected surfaces. The command is disabled if there are not two or more 3D surface plots open. If less than two surfaces are selected this command does nothing.

The X and Y coordinates of the generated surface will match those of the first surface selected. If a given point is outside the extents of other surfaces, then that surface does not contribute to the average.

### 4.5.21 Switch Independent Variable command

This procedure (optionally) creates a new document in which the amplitudes from one curve in the active document are substituted for the independent variable (X values) for all other curves. This is especially useful for finding relationships between two or more measured phenomenon in a time series.

The curve whose amplitudes you want to use as the independent variable must have monotonically increasing X values, otherwise the relationship you are trying to determine might be misleading. If necessary (and appropriate) you can first sort the X values using the Sort command on the Edit menu.

DPlot will interpolate, if necessary, to find the amplitude of this curve at each X value for all other curves. If a given X value is outside the extents of that curve, this point is deleted.

After performing this operation on all curves, all new curves are sorted in increasing X order if the **Sort new curves in new X order** box is checked.

If the source curve had a legend, it will be used as the X axis label for the resulting plot. Otherwise the Y axis label will be used.

You can call this function from a macro or via DDE conversation from another program with the <u>SwitchX</u> command.

### 烤 Related macro commands

SwitchX

### 4.5.22 Plugin modules

Shown below is a description of the <u>Generate</u> plugin modules distributed with the licensed version of DPlot.

Please note the effect of <u>Amplitude Limits</u>: For all Generate plugins which produce new data within the same document (as opposed to replacing existing data or producing a new document), DPlot sends only the data that lies within those limits to the plugin, so points outside those limits are effectively ignored. This applies to all plugins distributed with DPlot version 2.3.2.3 or later and to all <u>Version 4</u> plugins distributed by others. If this manipulation is undesirable (that is, you **want** to include points outside the limits), turn off *Amplitude Limits* before using these functions. For plugins that create a new document, **all** points are considered and the *Amplitude Limits* settings are applied to the new document. Note that this does not necessarily mean that points outside those limits are ignored, and they may effect the result depending on exactly what the plugin does. If you do not want any possibility of those points effecting the product, first delete them with the <u>Clip Y</u> command on the Edit menu.

### Average Curve with Error Bars

Produces a new document with amplitudes equal to the mean of all Y values within a user-specified interval in X, and error bars with extents equal to either:

- the mean minus the extreme values,
- · the standard error for the points within the given interval, or
- the standard deviation for the points within the given interval.

As noted on the input screen, this function operates on individual curves only. If you want to operate on **all** curves producing a single resulting curve, then exit this dialog, select "Combine Curves" on the Edit menu, select all curves, uncheck "Keep originals", click OK, then return to this function.

### **Backbone Curve**

Finds the points with maximum Y values within a specified interval on the X axis.

### **Baseline Shift**

This routine differs from the <u>Edit menu command</u> of the same name in that it will automatically select the time at which the integral of the record should be zero. For explosive events this generally means finding the arrival time of the shock. This is a big time-saver for those involved in data processing, particularly if used in a macro with the <u>RunPlugin</u> command. The procedure is described in detail on the <u>Baseline Shift</u> web page.

### Make Step Plot

Doubles the number of data points in a curve to give a stepped appearance. The original data will be replaced.



### **Monthly Peak**

Useful for finding the peak monthly value within a time series (cumulative daily sales figures, for example).

### More Curve Fits

This plugin is quite a bit more general than the least squares polynomial curve fits provided with the trial version of DPlot. This command allows you to fix one or more coefficients in the equation at a constant value, and also provides many more equation forms than a simple polynomial.

### **Moving Average**

Choose from prior (identical to Excel's moving average feature), central, or cumulative. For more information see the <u>blog entry</u>.

Starting with DPlot version 2.3.2.1, if using this plugin with a <u>RunPlugin</u> command, you may specify options for that command and bypass the input dialog.

Syntax:

[RunPlugin("Moving Average", "Curve number, Interval, Method, Tails")] where

Curve number = 1-based curve index to operate on.

Method = 0 for prior moving average, 1 for central moving average, or 2 for cumulative moving average

Tails = non-zero to include data tails outside the limits of the interval

In an Excel VBA macro "[RunPlugin(""Moving Average"", ""1, 12, 0, 1"")]" will use the "prior" method on curve number 1 with an interval of 12 and include the tails (all points including the first 11).

### **Poincare Plot**

Produces a new plot with X equal to the input Y and Y set to the next Y value in the curve. This operation is used to quantify self-similarity in processes.

### Rank

Similar to Excel's RANK function, but the output (X) is the percentage of amplitudes in the input that are less than the input Y. Produces a new plot with Rank on the X axis and the original Y values left intact.

### Split at Y=

Makes a copy of the input curve and sets values below (on the input) or above (on the copy) to userspecified Y values below or above these <u>Amplitude Limits</u>. One use of this feature is to subsequently use 2 different colors, line styles, or symbols on what was previously the same curve.

# 4.6 Info Menu

### **List Peak Values**

Displays a dialog box listing minimum and maximum amplitudes and the point at which they occur, along with mean, standard deviation, and standard error for each curve. **Shortcut:** 

-

Click on the toolbar.

### List Peaks within a Range

Identical to **List Peak Values**, but first allows you to specify a range in X or an X,Y window for 3D/4D data using the cursor. Click on the plot at one end/corner of the desired range, then drag the cursor to the other end/corner of the range and release to display the minimum and maximum amplitudes and other parameters within the selected range. For 3D and 4D surface plots this command is only available for 2D plots.

### Shortcut:

Click 🔟 on the toolbar.

### Interpolate...

Reports an interpolated amplitude for each curve at a user-specified abscissa.

### Scan X, Y

Displays data values for a single curve as you scroll through the data.

### Find Baseline

Finds the average amplitude between a user-specified range of points. Useful for performing baseline shifts.

### Number of Points

Reports the number of data points in each curve.

### **Find Intersections**

Finds the intersection(s) of one curve with another curve, optionally adding data points at the intersections and labelling those points. Both curves must have monotonically increasing X values. If necessary you can use the *Sort* command on the *Edit* menu prior to *Find Intersections* to sort curves in increasing X order.

### 烤 Related macro commands

### **FindIntersections**

### **Distance from**

Reports the distance and slope between two points (for XY plots, not necessarily points on a curve). You can use your mouse to click and drag from one point to another, or of course enter the coordinates in the respective text boxes. Please note: the distance reported is simply the square root of the sum of the squares of the horizontal and vertical offsets, in whatever units are used for X, Y, and Z for 3D/4D plots. The reported distance may therefore be meaningless. Exception: On XY plots and 3D plots viewed in 2D if both X and Y use either *Degrees, Minutes* or *Degrees, Minutes, Seconds* number format then DPlot assumes those values are longitudes and latitudes on the Earth, and reports the distance between points in miles, kilometers, and nautical miles.

For XY plots check "Snap cursor to nearest data point" to find the distance between two data points, rather than translated mouse position.

For 3D and 4D data, the cursor will always snap to the closest data point. In 3D views it is possible to click on a point that is hidden by the surface or by other data points in the case of a scatter plot. If this is not desirable you may need to use a bit of care when clicking.

# Shortcut: Click On the toolbar.

# 4.6.1 Interpolate command

### Shortcut:



Interpolates between data points and reports an interpolated amplitude at a user-specified abscissa. The method of interpolation is dependent on the type of scaling in use. On a linear scale, linear interpolation is used. On a logarithmic scale, linear interpolation is performed on the logarithms of the data points, and the inverse logarithm of the interpolated value is reported.

Note that you may get wildly different answers between one scaling type and another, particularly if the data points are widely spaced. An example of this apparent discrepancy is illustrated by the following two figures.





### 4.6.2 Scan X, Y command

### Shortcut:

) Click 🎴 on the toolbar.

Displays data values for a single curve as you scroll through the data. Unlike many other DPlot features, this command relies on the abscissa (X) values increasing monotonically; i.e. the data can be represented by a single-valued function. With curves that loop back on themselves (moment-thrust interaction diagrams, for example), you may not be able to scan through all visible points.

The Scan dialog box will report values that lie between the plot window's minimum and maximum X values, including values that lie either above or below the plot window.

Clicking on the scroll bar's right arrow, or pressing the <Right> or <Down> arrow keys, increments the index in the array by the value specified in the **Increment** box. Clicking on the scroll bar's left arrow, or pressing the <Left> or <Up> arrow keys, decrements the index. In both cases, pressing the Shift key increments the index by 10x the normal amount. The increment (initially 1) may be increased or decreased by a factor of 10 by clicking on the spin button arrows.

Clicking on the area to the left or right of the scroll bar "thumb", or pressing the <PgUp> or <PgDn> keys, moves the pointer by 5% of the number of points lying between the first and last points visible on the screen.

It may be necessary to zoom in to an area of interest to force the scroll bar to behave in a useful manner, particularly if the data record contains a very large number of points.

Click the **Copy** button to copy the X and Y values of the currently select data point to the clipboard using the default number formatting.

### 烤 Related macro commands

InfoScan

See also: View Menu

### 4.6.3 List Peaks command

### Shortcut:

Click 🖾 on the toolbar.

Displays a dialog box listing minimum and maximum amplitudes and the point at which they occur, along with other statistical parameters. Use the *List Peaks within a Range* menu command if you want these same parameters for a subset of the data.

If N is the number of points and y is the amplitude of each point, then the mean is given by

$$\overline{y} = \frac{\sum y}{N}$$

Standard deviation is given by

$$\sigma = \sqrt{\frac{\sum (y - \bar{y})^2}{N - 1}}$$

Standard error is given by

$$\sqrt{\left[\frac{1}{N\times(N-2)}\right]} \times \left[N\sum y^2 - \left(\sum y\right)^2 - \frac{\left[N\sum xy - \left(\sum x\right)\left(\sum y\right)\right]^2}{N\sum x^2 - \left(\sum x\right)^2}\right]$$

Standard error is only applicable to XY plots.

Skewness is a measure of the lack of symmetry in a distribution. Skewness is given by

$$\frac{\sum (y - \overline{y})^3}{N \times \sigma^3}$$

For N=1, skewness is reported as 0.

Kurtosis is a measure of whether a distribution has a high peak or is flat relative to a normal distribution. Kurtosis as calculated by DPlot is given by

$$\frac{\sum (y-\bar{y})^4}{N\times\sigma^4} - 3$$

For N=1, kurtosis is reported as 0.

This is also known as *excess kurtosis*. Three is subtracted from the formula as an adjustment to make the kurtosis of a normally distributed variable equal to 0.

# 4.7 View Menu

### All

Display the full plot extents, dependent on whether Manual Scaling is in effect.

Shortcut:

CTRL+1

### Zoom In

Allows you to specify the extents of the plot using the mouse. For polar plots, drag the mouse to define the radius. For all other supported graph types, drag the mouse to define a rectangle, or click and release the mouse on a point to zoom in by a factor of 2. Press the ALT key while dragging to force the zoom rectangle to be square.

Restrict zooming to the X direction by first pressing and holding the CTRL key. Restrict zooming to the Y direction by first pressing and holding the SHIFT key.

**NOTE:** With a linear or logarithmic scale or a Mercator projection, you can pan up, down, left and right by using your arrow keys or by clicking and dragging the mouse (if <u>Click and drag to zoom</u> is unchecked). When using the mouse, you can restrict the panning motion to horizontal, vertical, or 45 degree increments by simultaneously pressing the Shift key.

Arrow keys shift the plot by 1/2 the window width or height. Press Ctrl+arrow key to shift by 1/20 of the window width or height.

### Shortcuts:

Click



on the toolbar.

You can also zoom in by a factor of 2 by rotating the mouse wheel forward, and undo previous zooms by rotating the mouse wheel backward. When zooming in, new extents are based on the current mouse position. If the mouse is not within the plot, the new extents are based on the center of the plot. You can restrict zooming to the X direction by simultaneously pressing the CTRL key, or to the Y direction by pressing the SHIFT key. If you find that you often unintentionally zoom in or out by accidentally rotating your mouse wheel, you can disable this feature with the **Disable zooming with mouse wheel** checkbox under the General Options menu command.

If <u>Click and drag to zoom</u> on the General Options screen is checked, you can zoom by clicking and dragging without first clicking the **Zoom** toolbar button. (If this option is unchecked, clicking and dragging pans the plot.)

Ctrl+2, Ctrl+3, Ctrl+4, and Ctrl+5 zoom in by a factor of 2, 3, 4, and 5, respectively, on the current mouse position. If the mouse is not within the plot, the new extents are based on the center of the plot.

### 烤 Related macro commands

**SetPanFraction** 

### Zoom Out

Displays the previous view (before the last Zoom In command) of a plot.

Shortcut: Click

on the toolbar.

### Hide curves...

Allows you to hide one or more curves on an XY plot, which might be useful with multiple curves. Alternatively, if the document has a legend you may right-click on the legend entry for a curve and select "Hide all but <label>" to hide all curves other than the selected curve, or "Hide <label>" to hide the selected curve.

### 💋 Related macro commands

HideCurve ShowCurve

### Unhide all curves

Makes all curves visible (turns off the invisible flag set with "Hide all curves but...")

### **Reorder Curves/Groups**

Allows you to shift a curve (XY plots) or group (1D plots) up or down in the list of curves/groups. Alternatively, for XY plots if the plot has a legend you may right-click on the legend entry for the curve you want to move and select "Move up" or "Move down".

### Related macro commands

ReorderCurves

### Keep DPlot on top

If checked, the DPlot window will remain on top of all other windows, even when another application's window has the input focus.

### Status Bar

Toggles the display of the status bar at the bottom edge of the window.

### 烤 Related macro commands

ViewStatusBar

### Toolbar

Toggles the display of the toolbar. See Toolbar Description

### 烤 Related macro commands

**ViewToolBar** 

### Large Buttons

Toggles the size of the toolbar buttons between the default 24x22 pixels and 32x30 pixels.

### 32-bit Toolbar Icons

Check this item to display 32-bit toolbar icons. Uncheck this item to use 4-bit images. This command is disabled on Windows 2000 and earlier versions, as well as on systems set to display 256 or fewer colors.

### **Crosshair Cursor**

Toggles the display of crosshairs in place of the normal cursor. This feature is useful for visually comparing the amplitude of a point to a reference value.

Shortcut: Click on the toolbar.

### Restrict mouse panning to H/V/45°

If checked, panning a 2D plot by dragging the mouse is restricted to horizontal, vertical, or 45° increments. Normally this restriction requires pressing the Shift key while dragging.

### **Playback**

Allows you to animate XY plots.

### Acceleration (or Velocity/Displacement)

Toggles the view in Data Processing mode for data types that produce multiple plots. This menu command is hidden if Data Processing mode is not in effect.

### See also:

Data Processing Mode

Status Bar Description

### 4.7.1 Playback command

This command allows you to animate XY plots, drawing the plot data at a rate that you specify. Unlike the majority of DPlot's menu commands, the Playback command operates simultaneously on **all** open document windows that contain XY plots, with the limitations mentioned below. DPlot updates each plot drawing 30 times per second.

### Rate

Time scaling used for the playback. If the X axis label contains "msec" or "millisec" DPlot assumes that the X axis units are milliseconds. Otherwise seconds are assumed: 1 unit on the X scale = 1 second. The actual playback duration will be the X extents of the data in the assumed units divided by the **rate** entry. For example, if the X axis label is "Time, milliseconds" and the extents of the plot are 1500 units (1.5 seconds), a **rate** of 0.1 gives a playback duration of 15 seconds.

### Limitations

This feature will only work on documents that use linear scaling on both the X and Y axes. There is no harm in having an open document that does not use linear scaling, but that document will not be animated.

Curves are drawn using the color palette for the currently active document, regardless of palette settings for other open documents. Symbols are **not** drawn. Curves are drawn using solid lines with the widths specified in the active document. Performance will benefit if you set the line widths to 1 pixel for all curves before using this command. You can force single pixel lines by setting the line widths to 0" on the <u>Symbol/Line Styles</u> dialog. The plot will be redrawn using the specified line style, color, and width when the playback dialog is closed.

The performance of this feature is of course dependent on the speed of your processor and the total number of points to draw in any one 1/30th second frame. If the playback jumps it may be necessary to decrease the **rate** entry. By way of example, a 1.44Ghz processor should easily handle up to 8 200Khz records played back at up to 20% of the data rate (40000 points per second x 8 curves). A P300, on the other hand, will choke on this load and the drawing will not be able to keep up with the desired framerate.

### Start time

For multiple documents, DPlot will start redrawing each document so that the time scales are synchronized. For example, if **rate**=0.5 and document A starts at 0 seconds and document B starts at 3 seconds, the first line segment in document B will be drawn 6 seconds after the first line segment for document A.

### Video playback

If you have an *audio-video interleaved* (AVI) file associated with your data, you can display the video frame-by-frame synchronized with the display of your data. Click the video button to select an AVI file. You will then be presented with an **AVI Playback** dialog with these entries:

### File playback rate

This is the default framerate for the AVI file. This framerate is used by typical multimedia applications when playing back your file.

### Actual recording rate

Enter the framerate used when your video was recorded. On entry this value is set to the same value used the last time you used this feature, or to the file playback rate the first time you use this function. Typically, explosive events are recorded at a very high framerate (2000-5000 fps), but then stored as an AVI file at a much slower rate, typically 30-60 fps. Unfortunately this

information is **not** stored in the AVI file. You must be able to supply this value to have any hope of synchronizing the video playback with the rendering of your data.

### First frame at t=

Enter the time value for the first video frame. When playing back your data, the video playback starts at this time. The preview window showing individual frames of the video should assist you in determining this value. **Note:** Some video formats, DivX and MPEG4 in particular, will be very sluggish on most systems when scrolling backward through the video. This is an inherent problem associated with the compression method rather than a problem with DPlot or AVIFIL32.DLL.

After accepting your input by clicking OK, a window showing the first video frame will be opened in the lower right corner of your screen. Although this window has no caption, you may move it by clicking anywhere within the window and dragging to the desired location.

# Step forward, step back

If a video is used, the step forward and back buttons will advance/back up by one video frame. If no video is in use, the time step for each forward/back click is roughly equal to the extents of all plots divided by 100.

### Video Example

You have several document windows open which depict pressure measurements from an explosive event. X=0 corresponds to the detonation. Your video was recorded at 2000 fps, and the detonation occurs in the 19th frame of the video. Eighteen frames before the detonation is 18/2000 = 0.009 seconds. Enter "2000" for **file playback rate** and "-0.009" for the **first frame at t=** entry.

### Limitations

This feature uses your Windows' AVIFIL32.DLL to interpret AVI files. Although your system may have the appropriate codec to display a particular file format, that format may not be supported by AVIFIL32. This is a Windows version-dependent limitation. For example, most Windows 95 systems will not be able to display AVI's with MPEG4 compression, while Windows 2000 and XP systems will.

The limitations mentioned above concerning data rate are of course also applicable here. If playing back the video results in poor performance with rendering your data, it may be necessary to use a smaller **rate** factor.

DPlot updates both the document windows and the video display 30 times per second, regardless of the video framerate. Depending on the values you've entered for **actual recording rate** and **rate**, this may result in video frames being skipped. If this is the case and it is important to view every video frame, use a smaller **rate** entry.

### 4.7.2 Toolbar description

The toolbar contains buttons that give you quick mouse access to many commands and features in DPlot. For example, you can open a new document by clicking the New button on the toolbar.



New button

You can display or hide the toolbar by choosing Toolbar from the View menu. You may need to enlarge the DPlot window to the full width of the screen to see the entire toolbar.

You can see a description of each toolbar button in the status bar. When you point to any button with the mouse, the button description will appear in the status bar. A short description of each toolbar button is also provided in a ToolTip. When you point to a button with the mouse, the button name will appear in a box just below the mouse cursor:



For a detailed description of the actions taken by any of the toolbar buttons, click on the Help button, then click on the button you'd like more information about.

Help button

See also:

Moving the toolbar

# 4.8 Window Menu

### Cascade

Displays the open document windows in an overlapped arrangement, with the active document on top.

### Shortcut:

SHIFT+F5

### Tile

Arranges the open document windows as non-overlapped windows. The windows will be sized with a preference for tall rather than wide.

Shortcut:

shift+f4 🍉

### **Tile Horizontally**

Identical to **Tile**, but the windows will be sized with a preference for wide rather than tall. **Shortcut:** 

SHIFT+F6

### Arrange Icons

Arranges any minimized document window icons in the lower left portion of the DPlot application window.

### Related macro commands

WindowCascade WindowTile



### 5 **Programmer's Reference**

Using WinExec to plot an existing file Sending data to DPlot from another application Plugins for File Import, Export and Data Manipulation

### 5.1 Using WinExec to plot an existing file

The following code fragment (in FORTRAN) demonstrates a method for instructing DPlot to plot an existing file from within another Windows-based program.

```
! Handle of DPlot window
      integer*4 hWndDPlot
      integer*4 hInstDPlot
                                 ! Instance handle of DPlot module
      integer*4 wStatus
                                 ! API return code
      integer*4 hKey
                                 ! Registration database key
      integer*4 dwLength
                                 ! String length
      character szFileName*80
                                 ! Null-terminated string containing
                                  ! name of file to plot.
      character szDPlot*256
                                  ! Full d:\path\filename.ext
                                  ! of DPLOT.EXE
     hWndDPlot=FindWindow('DPlot'c, NULL POINTER)
      if (hWndDPlot.eq.0) then
        DPlot is not currently running. Check user's
         registration database for location.
С
        wStatus = RegOpenKey( HKEY_CURRENT_USER,
     1
                               'Software\DPlot\shell\open\command'c,
     2
                               hKey )
         if( wStatus .eq. ERROR_SUCCESS ) then
            DPlot's location is in the
            registration database... now retrieve it.
            dwLength=len(szDPlot)
            wStatus=RegQueryValue( hKey, NULL_POINTER,
     1
                                  szDPlot, dwLength )
            wStatus=RegCloseKey( hKey )
         else
            szDPlot = 'DPLOT.EXE'//char(0)
         endif
      else
```

С

С

С

C

С С

С С

DPlot IS running. Use API routines to get filespec.

```
hInstDPlot=GetWindowWord(hWndDPlot,GWW HINSTANCE)
   wStatus=GetModuleFileName(hInstDPlot,szDPlot,
1
                              len(szDPlot))
endif
dwLength=index(szDPlot,char(0))
wStatus = WinExec(szDPlot(1:dwLength-1)//' '//
1
                   szFileName,SW SHOWNORMAL)
if( wStatus .lt. 32 )
   wStatus=MessageBox(hWnd, 'Cannot find DPLOT.EXE'//char(0),
1
2
                       'MYAPPLICATION'//char(0),
                       MB OK .or. MB ICONEXCLAMATION )
3
```

Notes:

Registration database. DPlot's installation program automatically changes the user's registration database to indicate the location of DPLOT.EXE. DPlot also checks the entry in the registration database against its location every time it runs, and modifies the database entry if necessary. (This might be necessary if the user has moved the program since installing it.)
 WinExec. DPLOT.EXE takes a substantial amount of time to load the first time on slower

systems. However, if the user does not close DPlot, subsequent calls to WinExec are relatively fast.

# 5.2 Sending data to DPlot from another application

Beyond Copy/Paste and writing/reading files, there are 3 methods available for sending data to DPlot via dynamic-data exchange (DDE):

- The DPLOTLIB DLL hides the complexities of DDE and provides a simple interface for sending command strings to DPlot from a variety of programming environments. Source examples for Microsoft C, C#, Borland C++, LCC-WIN32, GNU C (gcc), Microsoft Visual Basic, Visual Basic . NET, PowerBasic, FreeBASIC, Absoft FORTRAN, Compaq Visual FORTRAN, GNU FORTRAN, Salford FORTRAN, WATCOM FORTRAN, and Delphi are included in the DLL distribution. The DLL, example programs, and source for the examples may be found in or below the \DPLOTLIB folder of the destination folder for DPlot.
- 2. For Excel users only, the optional <u>DPlot Add-In</u> will handle most of your needs. For everyday users it is as simple as selecting the data you want to plot, then selecting the appropriate Add-In command. For VBA programmers you can make use of Add-In commands to quickly produce a plot and then manipulate other plot parameters: line weight/style, symbol height/weight, colors, etc. See the <u>VBA programming</u> section of "Using the DPlot Interface Add-In for Microsoft Excel" for more information.
- 3. You can send a command or series of commands to DPlot from another application through a dynamic-data exchange (DDE) conversation, or "channel." For commercial applications, you should refer to the documentation for the application to determine the method for initiating a conversation with DPlot and executing commands for another application.

This method is **not** recommended unless you are very comfortable with DDE **and** you need more control over DPlot than DPLOTLIB.DLL provides. (In the latter case, contact us to let us know exactly what you are trying to do. It is likely that there is already a better way.)

Following is a brief demonstration of how to send a command to DPlot from three different environments.

### Visual Basic 3.0

Control.LinkMode = 0 Control.LinkTopic = "DPlot|System" Control.LinkMode = 2 Control.LinkExecute "[FileNew()]" Control.LinkMode = 0 where Control is any label, picture box, or text box.

### Microsoft Excel 4 macro

Channel =INITIATE("DPlot","System")

=EXECUTE(Channel,"[FileNew()]")

=TERMINATE(Channel)

### Microsoft Office products Visual Basic

Channel = DDEInitiate("DPlot", "System") DDEExecute Channel, "[FileNew()]" DDETerminate Channel

Following is a list of the commands supported by DPlot. All commands must be enclosed by square brackets. There should be no spaces in the command string, other than those embedded in character string arguments or used to delineate numeric parameters. Character arguments are always enclosed by double-quotation marks. Command names are not case-sensitive.

Some commands are valid only in macros (noted by <u>Macros Only</u>). Commands sent to DPlot via DDE must be enclosed by square brackets [ ]. Macro commands should **not** include the brackets.

Command parameters shown in the descriptions are placeholders for the actual values. Command parameters are either numeric values or character strings; character string parameters are always bound by double quotation marks.

All indices into arrays are 1-based, e.g. Curve1=1 refers to the first curve in a plot.

A Ox prefix for numbers in the descriptions below indicates hexadecimal notation; e.g. 0x0010 = 16.

✓ JR/Viewer indicates that the command is supported by DPlot Jr or DPlot Viewer.
 × JR/Viewer indicates that the command is NOT supported by DPlot Jr or DPlot Viewer.

**AAxisLabel** About Activate **AddCurves** AppHide AppMaximize **AppMinimize** AppRestore **AppShow AutoScale AutoSize** AverageAll **AverageAndReplace** Axes_At_Zero **AxisArrowheads** AxisColor

AxisType **□ B BarChartStyle BarWidth BaselineShift BkColor BkPlotColor Bubbles □** C Caption **ClearEditFlag** ClipY Color **Color Basis** ColorCycle **ColumnsAre CombineCurves** Compare Contour3D **Contour Axes ContourBarWidth ContourColorScheme ContourCustomColors ContourCustomLevels ContourDrawSides ContourGrid ContourLabelLines** ContourLegend **ContourLegendFrequency ContourLevels** ContourLighting **ContourMethod ContourResetExtents Contour Scales ContourView ContourViewChange** CopyPicture CopyText CurveFit CurveLabel 🗉 D **DataProcess DataProcessLabels** 

DateFormat **DecimalDigits DeferMinMaxCheck** Deletelmage **DeleteLabel DeleteNote** DeleteRefLine **DeleteThinTriangles DenseGridLines** Differentiate **Directory DivideCurves** 

**DocHide DocMaximize DocMinimize DocRestore DocShow** вE **EditAddToX EditAddToY** EditDecimate **EditEqualInts EditErase EditEraseData** EditFilter **EditMultiplyX** EditMultiplyY **EditNormalize EditOperateX EditOperateY EditOperateZ EditRemoveSpikes** EditRemoveTrend **EditReplaceData EditSmooth** EditSort EditSubtractMean **EditSwapXY EditTruncate** Enable ErrorBar ErrorBarPenWidth **ErrorBarWidth ErrorMessages** ⊫ F FFT **FFTMethod FFTPhase FileAppend FileArrays FileClose** FileExit **FileNew FileNewEx FileOpen FilePageSetup FilePrint** FilePrintMultiple **FileSaveAs FileType** FillBetween **FillStyle FindIntersections FontPoints** FunctionOfXY1Y2 FXY1Y2_XFrom

**GeneralOptions** GenerateMesh **GetPreferences** GrainSizeFlags GrainSizeFlagsEx **GrainSizeOther GridLines** GridLineType GridLinesOnTop ⊟ H **HelpContents HelpSearch HideCurve Histogram** -**IndependentTicks** InfoInterpolate InfoScan InsertImage **InsertLink InsertXY** Integrate **InvertAxis** ΒL LabelDirection Legend **LegendParams** LegendShow LineType **LineWidth ListPeaksOptions** LogTicks Lump **■** M **ManualScale** Medians **MenuHide** MenuShow **MenuUpdates MoveCurves MultipleXAxes MultipleYAxes MultiplyCurves** 🗉 N **NoteButtonText NoteOptions NotePosition NoteTabstops NoteText NSEW** NumberFormat **NumTicks OverlayDocument**  B P **Paste** PasteLink PolarPlotStyle **PostponeRedraw** Preserve3DScaleFactor **ProbabilityFlags □ R** Refline RemoveLink Request **ReorderCurves** Repaint **RunPlugin ■** S **SavePreferences** SaveUndo Scale **ScatterPlotLines SelectCurve** SetBackgroundImage SetBitmapDims **SetClientRect SetDecimalSymbol SetFilename** SetImageCrop SetMetafileDims SetOutputWindow **SetPanFraction SetPluginImageDims SetWindowPos ShowCaption** ShowCurve **ShowDate** <u>Size</u> **SplitDateTime** SquareLogCycles **StartStop** Stat_AddData Stat_AddGroup Stat_Box Stat BWOptions Stat DotOptions Stat_DrawLegend Stat GrandMean Stat_GroupInfo Stat_LimitsAsCL Stat MeanFormat Stat Outliers Stat_PlotType Stat_RefLine Stat_Rotated Stat_Scaling Stat_Whiskers SubtractCurves

**SwitchX SymbolClip** SymbolCycleColors SymbolFillBackground **SymbolFrequency SymbolOpacity** SymbolOutline SymbolOutlineColor **SymbolSize** SymbolSizeUnits SymbolType ΞT **TextFont TextNote** TextNote3D **TextNoteEx TextNoteLeader TextPointLabel** TextPointLabel3D **TextPointLabelLeader TickInterval TickMarks TimeFormat** Title1 Title2 **Title3** Title4 **TriangleFlags TripartiteFlags □ U UseNameAsLegend ■ V VectorPlot VectorPlotOptions** ViewRedraw **ViewStatusBar ViewToolBar − W** WindowCascade **WindowTile ■** X **XAxisLabel** X2AxisLabel XLabel XY **XYLabel** ΧΥΧΥ XYZ XYZ_to_XY XYZEx **XYZRegen □** Y **Y2AxisLabel** Y2TopDown

```
YAxisHorizontal
YAxisLabel
YCutoff

Z
ZAxisLabel
```

Commands may be issued in sequence or appended to each other. For example, the following command string opens and prints a DPlot file entitled "TEST.GRF" before causing DPlot to close:

### [FileOpen("TEST.GRF")][FilePrint()][FileExit()]

### Notes

1) Each DDE conversation has an association with a DPlot document. This association is established when the conversation is initiated, and may be changed by an <u>Activate</u>, <u>FileOpen</u>, or <u>FileNew</u> command. If, for whatever reason, either DPlot or the client application terminates the conversation and the client then establishes a new conversation, this association no longer holds. The client application should re-establish the association with a document with the <u>Activate</u> command, else any commands sent to DPlot may not operate on the intended document.

2) Character string arguments are always enclosed by double quotation marks. In some environments (Visual Basic, for example), double quotation marks are also used to delineate all character strings (including the command itself). The following example will always cause a syntax error in Visual Basic:

ret = DPlot_Command(docnum,"[FileOpen("myfile.grf")]")

Instead, use:

```
ret = DPlot_Command(docnum,"[FileOpen(""myfile.grf"")]")
```

in C, C++, C# you'd accomplish the same thing with:

ret = DPlot_Command(docnum,"[FileOpen(\"myfile.grf\")]");

3) You can use equations for numbers in the argument list to many commands. Equations must be prefaced by an equals sign (=). For example,

TickInterval(1,=10.^(FLOOR(LOG10(\$XRANGE)-1)))

sets the tick mark interval on the X axis to an integer power of 10 such that there will be at least 10 intervals but less than 100. For more information see the list of <u>available functions</u> and the <u>text</u> <u>placeholders</u> topics.

4) For real-time applications, it is strongly suggested that the client application use the <u>ManualScale</u> command to force the plot coordinates to the desired value. Otherwise, DPlot will continually adjust the plot coordinates to fit the incoming data. If the **ManualScale** command is issued DPlot keeps the extents of the X axis constant, shifting the plot window to the right when necessary. (If you do **not want** this shift to take place, use [<u>SetPanFraction(0)</u>].) For real-time applications that display a graph in their own window (rather than a DPlot document window), you will have much better performance using the <u>SetOutputWindow</u> command rather than retrieving a bitmap or metafile from DPlot after every graph update. For best performance turn off antialiasing with [GeneralOptions(32,0)].

5) For linear X, linear Y scales, DPlot draws individual line segments rather than redrawing the entire plot every time a new data point is added with the <u>XY</u> command. DPlot will only redraw the plot if the incoming data point lies outside the current plot extents. Although this optimization results in significantly faster plotting, there are several drawbacks: a) Since points are added one at a time rather than drawing a series of line segments, the pattern for a line style starts over at each point. Rather than using the slower line drawing routines that use a pattern (and would be incorrect anyway), DPlot uses the faster Windows API routines MoveTo and LineTo with solid lines. b) DPlot does not create and load the color palette selected by the user. Instead, the colors used to draw the line segments are the closest

match of the system colors to the desired colors. Both of these limitations apply only to line segments that are drawn individually. Once the entire plot is redrawn, the lines will be drawn correctly.

6) DPlot returns one of 3 values in response to any command. The return codes sent to the DDEML (Dynamic Data Exchange Management Library) are DDE FACK (=0x8000), DDE FBUSY (=0x4000), and DDE FNOTPROCESSED (=0). The various development environments respond to these return codes in different manners, usually by displaying a MessageBox in the case of DDE FBUSY or DDE FNOTPROCESSED. DDE FACK indicates success. DDE FBUSY indicates that DPlot is currently disabled (possibly because a MessageBox or other modal dialog box is displayed). DDE_FNOTPROCESSED indicates that the command was not valid or was unsuccessful for some other reason (exceeding the array sizes, for example). Note that DPlot does not cache incoming commands if it is currently busy... it is up to the client application to re-send the commands later, if necessary. Real-time client applications should not rely on DPlot to save incoming data values.

### For optimum performance with real-time applications

- 1) If a graph is being rendered in DPlot (the default) rather than the calling application, prevent the user from selecting a menu command during real-time processing by using the Enable command, and turn off menu updating with a MenuUpdates command.
- 2) Use the ManualScale command.
- 3) Set all line widths to 0 with the LineWidth command for older processors.
- 4) Set all line styles to solid (*type* = 1) with the LineType command.
- 5) If rendering a graph in a window owned by the calling application, use SetOutputWindow.
- 6) Turn off antialiasing with [GeneralOptions(32,0)].

#### 5.2.1 AAxisLabel macro command

### [AAxisLabel("string")]

### Sets the amplitude axis label used on 4D surface and scatter plots to string. Limited to 200 characters. This label will be drawn above the legend for surface plots and for scatter plots for which the "Color Based On" entry is set to "Amplitude". The text will be drawn using the Legend font. For formatting codes (**bold**, *italic*, subscripts, color, etc.) see the description of the Title/Axes menu command.

See also

Title/Axes menu command

#### 5.2.2 About command

[About()]

Displays the About dialog box for DPlot.

### 5.2.3 Activate command

[Activate("caption")]

### [Activate(DocumentNumber)]

Activates the document with the character string *caption* in the title bar or the document index DocumentNumber. Subsequent commands reference this document. The document number for the document window associated with a given DDE conversation may be retrieved from DPlot by sending an XTYP REQUEST (LinkRequest in Visual Basic jargon) on the item "DocumentNumber".

### ✓JR XViewer

✓JR √Viewer

✓JR √Viewer

### 5.2.4 AddCurves command

### [AddCurves(*Curve1,Curve2*)] XJR Viewer

Generates a new curve with amplitudes equal to the amplitudes of *Curve1* plus the amplitudes of *Curve 2*. For example,

[AddCurves(1,2)]

creates a new curve with amplitudes equal to the sum of the amplitudes of the first and second curves.

X values for the new curve are taken from *Curve1*, subject to whether <u>FXY1Y2_XFrom</u> with a nonzero *state* value has been previously used. Linear interpolation is used to find the amplitude of *Curve2* at each X.

✓JR √Viewer

See also

Y=f(X,Y1,Y2,...) menu command

### 5.2.5 AppHide command

[AppHic	<b>le()</b> ]	l				√JR	🗸 Viewer

Hides the DPlot application window.

### 5.2.6 AppMaximize command

[AppMaximize()]

Maximizes and activates the DPlot application window.

### 5.2.7 AppMinimize command

### 5.2.8 AppRestore command

[AppRestore()] ✓JR ✓Viewer Restores (to unmaximized or unminimized size) the DPlot application window.

### 5.2.9 AppShow command

### 5.2.10 AutoScale command

 on the Options menu.

### See also

Extents, Tick marks, Size menu command

### 5.2.11 AutoSize command

### [AutoSize()]

✓JR ×Viewer

XJR XViewer

XJR XViewer

Equivalent to unchecking the "Specify Plot Size" checkbox on the <u>Extents, Tick marks, Size</u> command on the Options menu.

### See also

Extents, Tick marks, Size menu command

### 5.2.12 AverageAll command

### [AverageAll()]

Creates a new curve consisting of the average amplitude of all curves in a document

See also

Average All Curves menu command

### 5.2.13 AverageAndReplace command

### [AverageAndReplace()]

Identical to AverageAll, but deletes all pre-existing curves after generating the average curve.

See also Average All Curves menu command

## 5.2.14 Axes_At_Zero command

### [Axes_At_Zero(state)]

Equivalent to the "Axes at 0" command on the Options menu. If **state** is non-zero, black lines with the same line weight as the coordinate axes will be drawn at X=0 and Y=0 if those values are within the extents of the plot.

✓JR XViewer

## 5.2.15 AxisArrowheads command

### [AxisArrowheads(state)]

Turns on (*state*=1) or off (*state*=0) arrowheads on the coordinate axes of XY plots when "<u>Axes Only</u>" is selected.

✓JR Xiewer

See also

AxisType macro command

### 5.2.16 AxisColor command

[AxisColor(red,green,blue)]

✓JR XViewer

Sets the color used for the coordinate axes. red, green, and blue should be between 0 and 255, inclusive. Black=0,0,0; white=255,255,255.

### 5.2.17 AxisType command

### [AxisType(type)]

### ✓JR XViewer

Equivalent to the Grid Lines or Box selections on the Options menu for XY plots. Use type=0 for Axes Only, type=1 for Grid Lines, type=2 for Box Around Curves, type=3 for Vert. Grid Lines Only, or type=4 for Horz. Grid Lines Only.

### See also

Grid Lines or Box Options menu command

### 5.2.18 BarChartStyle command

#### [BarChartStyle(flags)] ✓JR XViewer

Changes the appearance of bar charts. *Flags* is a combination of:

- 0x00000001 Filled bars (Obsolete, though supported. See FillStyle)
- 0x0000002 Stacked bars
- 0x0000004 Draw bar amplitude labels
- 0x0000008 Draw percentage labels
- 0x00000010 Labels drawn outside (on top of) the bars
- 0x00000020 For unevenly spaced X values, bars extend from X to the following X. Last point is ignored other than to provide the ending X of the last bar. Ignored if more than one data set is present.
- 0x00000040 Percentage labels (if drawn) are a percentage of this bar's amplitude relative to all data sets. If not set, percentage is for the associated data set only.
- 0x00000080 A different color is used for each bar. The color assigned to curve #1 (with the Colors command) is used for the first bar, and in general the color assigned to curve N is used for the Nth bar. This option is ignored if the bar chart has more than one data set.
- 0x00000100 Base of bars is at minimum Y extent. If not set, base of bars will be at Y=0.
- 0x00000200 Don't draw black border around filled bars.
- 0x00000400 Draw border around extremes only. In other words, if bar width and group width (see below) are both set to 1.0, there will be no vertical lines between bars. Ignored if more than one data set is present. If set, 0x0200 bit (don't draw black border) is ignored. 0x00000800 Shade bars as if they are cylindrical. Ignored unless the fill style is set to solid.
- 0x00001000 Horizontal bars.

Combine *flags* options by adding them.

# 5.2.19 BarWidth command

### [BarWidth(width,groupwidth)]

Controls the width of bars and optionally the space between groups of bars at the same ordinate (X value) in a bar chart. For width=1.0 there will be no space between adjacent bars at the same ordinate. Both width and groupwidth values are forced to the range 0.01 to 1.0, inclusive. To set the groupwidth value but leave the width value unchanged, use [BarWidth(,groupwidth)]

### 5.2.20 BaselineShift command

### [BaselineShift(time,value,start,stop)]

In data processing mode, modifies the primary record (pressure in a pressure-time history,

✓JR XViewer

XJR XViewer

acceleration in an acceleration-time history) so that the first integrated record (impulse or velocity, respectively) has an amplitude of *value* at *time*. Only shifts values of the primary record falling between the *start* and *stop* times. If *start* is a large negative number (< -1.E+19), the shift is applied to all amplitudes (the *stop* time is ignored). For more information see the description of the <u>Baseline</u> <u>Shift</u> menu command. As an alternative to this command, the Baseline Shift plugin (included with the licensed version) may be used to perform essentially the same task. The significant differences are that a) the plugin automatically determines the "time" value, b) "value" is always assumed to be 0, and c) it applies the shift to all points. To use the baseline shift plugin in a macro or DDE conversation, use [RunPlugin("Baseline Shift")]. This is a particularly handy tool when used in a ForFilesIn loop.

### 5.2.21 BkColor command

### [BkColor(color)]

### ✓JR XViewer

Sets the background color for the entire document window, including areas outside a plot. By default DPlot uses the Windows' window background color for the background of plots. The window background color may be set by right-clicking on your desktop and selecting *Properties*, then the *Appearance* tab, then the *Advanced* button. If *color* is -1, the window background color will be used. *color* is in the form (*blue**65536)+(*green**256)+*red*, where *blue*, *green*, and *red* range from 0 to 255. The *color* value may be expressed as a decimal or hexadecimal value; hexadecimal notation should include a "0x" prefix. For example 0x00FF00FF and 16711935 both represent magenta (*blue=red=*255, *green=*0).

**Please note:** Use caution with this setting. If used and *color* is **not** equal to -1, then the specified *color* will be used on **all** output, including printed plots. (By default the background is not painted on printed output.) A colored background will of course consume much more ink than a transparent background.

To set the background color for the area within a 2D plot, excluding areas outside the coordinate axes, see <u>BkPlotColor</u>.

### See also

Colors menu command

## 5.2.22 BkPlotColor command

### [BkPlotColor(color)]

### ✓JR ×Viewer

Sets the background color for a 2D plot. By default DPlot uses the Windows' window background color for the background of plots. The window background color may be set by right-clicking on your desktop and selecting *Properties*, then the *Appearance* tab, then the *Advanced* button. If *color* is -1, the document window background color will be used (set with <u>BkColor</u>). If that color value is also set to -1, then the window background color set with Control Panel is used. *color* is in the form (*blue* *65536)+(*green**256)+*red*, where *blue*, *green*, and *red* range from 0 to 255. The *color* value may be expressed as a decimal or hexadecimal value; hexadecimal notation should include a "0x" prefix. For example 0x00FF00FF and 16711935 both represent magenta (*blue=red=255*, *green=0*).

**Please note:** Use caution with this setting. If used and *color* is **not** equal to -1, then the specified *color* will be used on **all** output, including printed plots. (By default the background is not painted on printed output.) A colored background will of course consume much more ink than a transparent background.

To set the background color for the entire plot, including areas outside the coordinate axes, see <u>BkColor</u>. BkColor and BkPlotColor are not mutually exclusive: you can use BkColor to set the color of areas outside the coordinate axes and BkPlotColor for the area bound by the axes.

See also

Colors menu command

### 5.2.23 Bubbles command

[Bubbles(curve,symbol,source,size_lo, size_hi, color_lo,color_hi|,low,high,options|)] ✓JR XViewer

Sets bubble plot parameters for the curve with 1-based index *curve*. The symbol type is specified by the *symbol* parameter (see all available symbol types in the description of the <u>SymbolType</u> command). The symbol size will be based on the Y values of the curve with 1-based index *source*. The *source* curve will not be drawn. The X values for the *source* curve are ignored: the n'th point in *curve* will use the n'th point in *source*. The size of the symbols (*size_lo* and *size_hi*) is specified in thousandths of inches, i.e. 200 = 0.2 inches. The *size_lo* and *size_hi* sizes correspond either the minimum and maximum Y values in *source*, or to the *low* and *high* settings, dependent on *options*.

Colors are in the form blue*65536+green*256+red, where blue, green, and red are between 0 and 255, inclusive. Colors may be specified in hexadecimal notation with a '0x' prefix. For example 0x00FFFF00 is cyan (blue=255, green=255, red=0).

The optional *options* setting may be a combination of:

0x00000001 Solid symbols will be outlined in black

- 0x0000002 If set, bubble size/color scale is determined by the *low* and *high* values. If not set, size and color are determined by the minimum and maximum amplitudes in the *source* curve.
- 0x00000004 If set, bubbles with amplitudes outside the *low* to *high* range will not be drawn. If not set, those bubbles will be drawn with the size and color associated with the *low* or *high* limit. This flag is meaningless if the 2 bit is not also set.

The default value for *options* if it is omitted is 1.

Combine options by adding them.

Note: **source** must be different than **curve**. If you want to use a curve's amplitudes for the bubble values, make a copy of that curve and set **source** to that copy.

See also

Bubble Plot menu command

### 5.2.24 Call command

### Call macro Macros Only

✓JR √Viewer

✓JR √Viewer

Launches another instance of the macro editor and runs the specified macro. Control is not returned to the caller until the called macro has completed.

### 5.2.25 Caption command

### [Caption("caption")]

Sets the caption for the document referenced by the current DDE conversation or macro. The caption will be compared to all other open documents, and if it is not unique then a numeric suffix will be appended. If the plot does not already have an associated filename, the *caption* will be used along with an extension of ".grf" as the default filename for the plot. That filename will then be the default entry if the user elects to save the plot using the <u>Save As</u> command on the File menu.

### © 2014 HydeSoft Computing, LLC

# 5.2.26 ClipY command

# [ClipY(low,high)]

For an XY plot, deletes all points with amplitudes less than the *low* value or greater than the *high* value from the curve or curves specified with the last SelectCurve command. If no data remains as a result of this operation, the curve and all associated data are erased. If instead of deleting points you want to simply ignore points outside a range, see the YCutoff command.

XJR XViewer

# 5.2.27 ClearEditFlag command

### [ClearEditFlag()]

Clears an internal flag that indicates whether the document has been edited, so that, when closing the document, DPlot does not prompt you to save the document. Any subsequent changes to the document will reset the flag.

✓JR √Viewer

### 5.2.28 Color command

### [Color(index,r,g,b)]

Sets the color for the *index'th* curve of an XY plot to the color with red, green, and blue components r,g,b, where r,g, and b range from 0 to 255. If index = -1, then r,g,b are used for ALL curves.

[Color(3,255,0,255)] sets the color for the third curve to magenta, and

[Color(-1,0,0,0)]sets the color for all curves to black.

If index=0, r,g,b represent the color components for grid lines. For XY and 1D plots, the default grid line color is a light gray (192, 192, 192). For 3D plots the default grid line color is black (0, 0, 0).

For 3D/4D data only, if *index*=-2, *r,g,b* represent the color components for the border lines (lines drawn when "Borders" on the Contour Options dialog is checked.)

### See also

Colors menu command

## 5.2.29 ColorBasis command

### [ColorBasis(basis)]

Specifies the color-selection method used for 3D scatter plots. 0=Z value, 1=data set, 2=view depth

See also Scatter Plot Options menu command

## 5.2.30 ColorCycle command

### [ColorCycle(cycle)]

For XY plots, copies the first cycle colors to all subsequent groups of cycle colors. For example if cycle=10, then the color used for the first curve will be copied to the 11th, 21st, 31st,... colors, color #2 will be copied to the 12th, 22nd, 32nd,... colors, etc. This command only effects the current color

JR XViewer

### ✓JR XViewer

✓JR XViewer

specifications; it is not persistent. So if you subsequently edit color #3, for example, that change will not be propogated to color 3+*cycle*, 3+2**cycle*, etc.

### See also

Colors menu command

### 5.2.31 ColumnsAre command

### [ColumnsAre(*state*/,*col1*,*col2*,.../)] ×JR √Viewer

Determines how the data columns in a multiple column (type D or K, see thorough description in the <u>Open menu command</u> topic.) file are interpreted by a <u>FileOpen</u> command. Optionally allows you to specify which columns from the file are assigned to which arrays.

### state

0	XY plot. The first column is assumed to be X and all subsequent columns are assumed to be Y values, 1 curve per column, all sharing the same X. If <i>col1,col2,</i> are included then <i>col1</i> specifies the column for the X values.
1	XY plot. Columns are alternating X,Y values, 1 curve for every two columns.
2	XY plot. All columns are assumed to be Y values, 1 curve per column, with X starting at 0 and incremented by 1 for every row.
3	3D surface plot. Default column arrangement is X, Y.Z

The default *state* is 0 before a macro is run or a DDE conversation is initiated.

For *state*=0, 2, or 3, you may also specify which columns are assigned to X, Y, and/or Z.

For *state*=0, *col1* specifies the column for the X values and subsequent entries specify Y values, 1 curve per column, that share the same X. For example [ColumnsAre(0,2,1,4)] results in the 2nd column being used for the X axis, the 1st column being used for the Y values of the 1st curve, and the 4th column being used for the Y values for the 2nd curve.

For *state*=2, *col1* specifies the column for the Y values of the 1st curve, *col2* specifies the column for the Y values of the 2nd curve, etc. X for both curves will start at 0 and be incremented by 1 for every point.

For *state*=3, the column numbers should be listed in the order X,Y,Z. You **can** use the same columns for X and Z or Y and Z, but if the X and Y values are taken from the same column then the 3D "surface" will consist of colinear points and cannot be plotted.

### NOTE:

**state** alone does not determine what type of plot will be produced when opening a file. Multiple column text files are always assumed to hold X,Y data unless a <u>FileType</u> command precedes the <u>FileOpen</u> command. For this reason using **state**=3 but not also specifying the column arrangement is essentially meaningless.

The settings established by ColumnsAre are turned off after a FileOpen command is processed. So if you want the same settings to be used for multiple files, you must duplicate the ColumnsAre command before each FileOpen.

# 5.2.32 CombineCurves command

# [CombineCurves(*Curve1,Curve2,sort*], keep/ XJR XViewer

)]

Appends the X,Y points from *Curve1* to *Curve2*. If *sort* is non-zero, the resulting curve will be sorted in increasing X order. If the optional parameter *keep* is non-zero, the original curves will be preserved. Otherwise the original curves will be deleted.

### 5.2.33 Compare command

### [Compare(*Curve1,Curve2*)] XJR Viewer

Generates a new curve with amplitudes equal to the amplitudes of *Curve1* minus the amplitudes of *Curve 2*. This command has been kept for backward compatiblity and is equivalent to the <u>SubtractCurves</u> command.

### See also

Y=f(X,Y1,Y2,...) menu command

# 5.2.34 Contour3D command

### [Contour3D(state)]

Draws a contour plot with a 3D projection (*state*=1) or 2D view of the XY plane (*state*=0). Equivalent to checking (*state*=1) or unchecking (*state*=0) the *View in 3D* box on the <u>Contour Options</u> dialog box.

See also

Contour Options menu command

## 5.2.35 ContourAxes command

### [Contour Axes(state)]

Draws the coordinate axes of a 3D contour plot for *state*=1. Equivalent to checking (*state*=1) or unchecking (*state*=0) the *Axes* box on the <u>Contour Options</u> dialog box.

### See also

Contour Options menu command

## 5.2.36 ContourBarWidth command

### [ContourBarWidth(x,y)]

Sets the bar widths for 3D bar charts relative to the spacing between rows and columns in the X and Y directions. Default value = 1.0 for both. Forced to the range 0.0-1.0. This command is equivalent to the *Bar chart* settings on the <u>Contour Options</u> dialog box.

[ContourBarWidth(1,1)]

# ✓JR ×Viewer

√JR ×Viewer

✓JR XViewer

cos(x)*sin(y)



[ContourBarWidth(0.5,0.5)]

cos(x)*sin(y)



See also Contour Options menu command

# 5.2.37 ContourColorScheme command

### [ContourColorScheme(method)]

✓JR Xiewer

Specifies the color palette to use for contour plots. Uses the standard rainbow palette for **method**=0, a gray scale for **method**=1, or custom colors (which can be set with <u>ContourCustomColors</u>) for **method**=2.

### See also

ContourCustomColors macro command
### Contour Options menu command

## 5.2.38 ContourCustomColors command

# [ContourCustomColors(*color1,color2,...,* **JR X**Viewer *color[n-1],color[n]*)]

Sets up to 64 custom color values (blue*65536+green*256+red) to use for contour plots. **color1** is the color used for the low contour line/shade, **color[n]** is used for the high value. Colors may optionally be expressed in hexadecimal notation with a 0x prefix, e.g. 0x00FF0000 is blue. These colors are only used with [ContourColorScheme(2)].

### See also

Contour Options menu command

### 5.2.39 ContourCustomLevels command

### [ContourCustomLevels(z1,z2,z3,...,z[n-1],z[n])]

Sets up to 256 contour intervals for 3D or 4D surface or scatter plots. The values may be in any order and do not need to be sorted. This is equivalent to checking the *Custom levels* box on the <u>Contour Options</u> dialog box.

### See also

Contour Options menu command

### 5.2.40 ContourDrawSides command

### [ContourDrawSides(state/,color/)]

If **state**=1, draws planes from the edge of the 3D contour plot surface to the XY plane, obscuring the underside of the surface. Planes will be the color specified by **color** (blue*65536+green*256+red or 0x00bbggrr), outlined with the border color (set with [Color(-2,r,g,b)]). If **color** is omitted, planes will be white. Elevation angle (set with <u>ContourView</u>) less than 0 should **not** be used in this case. This command (for **state**=1) is equivalent to checking the *Sides* checkbox in the *3D* group on the <u>Contour Options</u> dialog box.

### Example:

[ContourDrawSides(1,0xFF0000)]

### ✓JR ×Viewer

✓JR XViewer



See also <u>Color</u> macro command <u>Contour Options</u> menu command

### 5.2.41 ContourGrid command

### [ContourGrid(state)]

✓JR XViewer

State is a combination of:

- 1 Draw lines through the contour control points. For random 3D points this will draw the borders of the triangles. For points on a rectangular grid this results in lines around each grid cell.
- 2 Draw grid lines across the plot at the same interval as the X and Y axis labels. The frequency of these tick marks can be changed with the Extents/Intervals /Size command on the Options menu.

See also

Contour Options menu command

### 5.2.42 ContourLabelLines command

### 

Specifies whether individual contour lines are labeled, the frequency of those labels, and formatting options for those labels. *flags* is a combination of:

0x00000001 Label contour lines. If not set, no labels are drawn.

- 0x0000002 Labels are transparent. If not set, labels are opaque and will obscure the underlying contour line. (This setting is ignored for "Both lines and shades" (<u>ContourMethod</u>(6)). For those renderings the labels will always be drawn opaque, with the background preserved.)
- 0x0000004 Labels are drawn in black. If not set, labels are drawn using the same color as the corresponding contour line. (This setting is ignored for "Both lines and shades" ( <u>ContourMethod(6)</u>). For those renderings the labels will always be drawn in black, subject to the 0x00000010 setting, see below.)
- 0x00000008 Labels will be drawn horizontally. If not set, labels will be oriented with the respective contour lines.

0x0000010 If the luminance (red*0.3 + green*0.59 + blue*0.11) of the region **above** the contour line is some threshold value or greater, the label is drawn in black. Otherwise the label is drawn in white. (This setting is only applicable when "Both lines and shades" ( [ContourMethod(6)]) is used.)

Combine *flags* options by adding them.

*frequency* specifies the frequency of the labels. If 0 or 1, every line is labeled; if=2, every other line is labeled, etc.

### See also

Contour Options menu command

### 5.2.43 ContourLegend command

## [ContourLegend(*state*/,*side*,*greatersymbol*/ ✓JR ×Viewer

)]

For non-zero *state*, draws a color-coded legend with values and intervals either automatically selected by DPlot or specified via the ContourLevels command. Equivalent to checking (*state* != 0) or unchecking (*state*=0) the *Legend* checkbox in the *Draw* box on the <u>Contour Options</u> dialog box.

The optional *side* parameter dictates which side of the plot the legend will be drawn on.

0 = left

1 = right

2 = top

3 = bottom

The default value is *side*=1 (right side), unless a settings file has been saved that includes the *side* setting.

The optional *greatersymbol* parameter determines whether DPlot will add  $\geq$  and/or  $\leq$  prefixes to the high and/or low contour legend entries. If non-zero and the extents of the amplitudes are more than one contour interval greater than the high contour value (or less than one contour interval less than the low contour value) then the corresponding prefix will be added to the legend. If set to 0, those prefixes will not be added. This setting has no effect on contour lines, only shaded bands and scatter plots.

See also

Contour Options menu command

### 5.2.44 ContourLegendFrequency command

### [ContourLegendFrequency(frequency)]

Controls how many (if any) legend entries are skipped on a contour plot. If set to 0 or 1, every label is drawn. If set to 2, every other label is skipped. If set to 3, every 3rd label is drawn, etc. Equivalent to the *Label every:* selection on the <u>Contour Options</u> dialog box.

See also

Contour Options menu command

### 5.2.45 ContourLevels command

[ContourLevels(numlevels,lo,hi)]

Allows you to specify the number of contour levels and the low and high extents for a contour plot. For unequal spacing between contour levels, see <u>ContourCustomLevels</u>.

### See also

Contour Options menu command

### 5.2.46 ContourLighting command

### 

Specifies how a 3D view of a 3D surface is lit. **method=**0 for no lighting (full brightness),=1 for Gouraud shading,=2 for Phong shading. **ambient**=ambient light fraction, 0 to 1 inclusive. Equivalent to the *Lighting* options on the <u>Contour Options</u> dialog box.

### See also

Contour Options menu command

### 5.2.47 ContourMethod command

### [ContourMethod(drawmethod)]

Sets the drawing type for 3D data to shaded bands for *drawmethod*=0, contour lines for *drawmethod*=1, waterfall plot for *drawmethod*=2, a wireframe mesh for *drawmethod*=3, a 3D bar chart for *drawmethod*=4, or a combination of shaded bands and contour lines for *drawmethod*=6. For waterfall, wireframe, and bar chart plots, <u>Contour3D</u> is forced to a 3D projection. For waterfall plots and bar charts, the data **must** be in the DATA_3D form (rectangular grid) rather than DATA_3DR (random 3D points). To produce an interpolated rectangular grid from random 3D points, use the <u>GenerateMesh</u> command.

✓JR XViewer

Contour lines (*drawmethod*=1) and the shaded bands/contour lines combo (*drawmethod*=6) are only valid for 2D views. If a 3D view is in effect when either of these options is selected, the plot type is forced to shaded bands (*drawmethod*=0).

### Examples:

[ContourMethod(0)]







[[]ContourMethod(3)]



[ContourMethod(4)]

```
cos(x)*sin(y)
```



To set the individual bar widths, use <u>ContourBarWidth</u>. [ContourMethod(6)] (2D views only)



### 5.2.48 ContourResetExtents command

### 

If *state* is non-zero then DPlot will automatically adjust the lower and upper contour limits when you zoom in/out and/or force the plot extents by using the <u>Extents/Intervals /Size</u> command on the Options menu. In this case the specified number of contour intervals set with <u>ContourLevels</u> will be used as a *guideline*; the actual number used may be slightly more or less. If you zoom in to such an extent that only one data point is visible, or if all visible data points have the same magnitude, the lower and upper limits will be forced to that value +/- 5%, or +/- 0.5 in the case of Z=0. A non-zero *state* is equivalent to checking the **Use actual displayed limits** box on the Contour Options dialog.

If *state* is 0, the specified lower and upper contour limits will be used regardless of the displayed extents.

See also Contour Options menu command

### 5.2.49 ContourScales command

### [ContourScales(*Xscale, Yscale, Zscale*)]

Sets the drawing scales for X, Y, and Z values on contour plots. This has no effect on the actual data values, only on appearance. By default the scale factors are 1:1:1 unless "Always auto-adjust scale factors on 3D plots" on the *General Options* dialog is checked. In the latter case, scale factors are automatically set such that the plot is square in the XY plane and the Z extents appear to be about 1/2 that of X and Y.

If any scale factor is set to 0, then the behavior depends on the "Always auto-adjust scale factors on 3D plots" setting. If checked (or set with a <u>GeneralOptions</u> command), then the corresponding factor is set to mimic that behavior without altering the other scale factors. If that option is **not** selected, a 0 scale factor is forced to 1.0.

A negative scale factor will reverse the normal direction for the associated axis. For example **[ContourScales(1,1,-1)]** will invert the Z axis so that the Z values increase from top to bottom in a 3D view.

### See also:

GeneralOptions macro command Contour Options menu command

### 5.2.50 ContourView command

### [ContourView(azimuth,elevation,perspective $\checkmark$ JR ×Viewer

)]

Sets the view angles for a 3D contour plot and the range from the viewpoint to the center of the surface for perspective projections. All arguments are optional and must be separated by a comma. Angles are specified in degrees. If *perspective* is not greater than 0, a normal orthographic projection is drawn (no perspective correction). To draw the plot in a 3D projection, use <u>Contour3D</u>. This command is equivalent to the *Azimuth, Elevation*, and *Perspective* settings on the <u>Contour</u> <u>Options</u> dialog box.

### Example:

[ContourView(,30)]

leaves the azimuth and perspective at their current values and sets the elevation angle to 30 degrees.

The *perspective* value is a multiplier for one-half the diagonal of the imaginary box surrounding the plot. Values greater than 0 and less than 1 are forced to the minimum of 1.0. A value of 1 places the viewpoint on the imaginary smallest sphere that encompasses the plot. Generally this value will produce a very distorted plot. Best results are usually achieved with a value of 2 or more.

### Perspective examples:

[ContourView(30,20,0)]



N

[ContourView(30,20,2)] N 3π/4 π/2 3π/4 π/2 π/4 0 π/2 π/4 r π/2 3π/4 3**π**/4 -7 [Contourview(30,20,3)] 12 π -// 3π/4 0 π/2 **π**/4 π/4 0 **π/2** -π/4 Y **π/2** 3π/4 -3π/4 -7 -π

# Contour Options menu command 5.2.51 ContourViewChange macro command

ContourViewChange macro command

### [ContourViewChange(delta_azimuth, ✓JR XViewer delta_elevation)]

Changes the 3D view angles by *delta_azimuth* and/or *delta_elevation*, in degrees. Both parameters should be in integer degrees.

See also ContourView macro command Contour Options menu command

See also:

## 5.2.52 CopyPicture command

### [CopyPicture()]

Copies metafile and bitmap representations of the document, as well as the current color palette, to the Windows clipboard.

### See also

Copy Picture menu command

### 5.2.53 CopyText command

### [CopyText()]

Copies the data values for the curve selected with the SelectCurve command to the Windows clipboard as text, one data pair per line. X and Y values are separated by [TAB] characters.

### 5.2.54 CurveFit command

### [CurveFit(order)]

Uses least-squares curve fitting procedures to find the **order'th** order polynomial that best fits the data values for the curve selected with the SelectCurve command. If successful, the CurveFit command generates a new curve using the fitting polynomial. The new curve will have the maximum number of points (set by the FileArrays command), equally spaced between the limits of the input curve. For more information, see the Help topic on the Polynomial Curve Fit menu command.

### 5.2.55 CurveLabel command

### [CurveLabel(curve, "string")]

Sets the label placed adjacent to the last data point for curve number curve to the character string string.

### 5.2.56 DataProcess command

### [DataProcess(state,"datatype",format)] XJR XViewer

Turns data processing mode on (state=1) or off (state=0). If state=1, datatype is a descriptive term for the data from the table below. Datatype must include enough characters to uniquely identify the type of data. For example, *datatype* for stress or strain measurements must include at least four characters. Format is 1 for wide plots, square format is used otherwise. Datatype and format may be omitted for state=0. For more information, see the Help topic on the Data Processing menu command.

### Data Types

Acceleration Deflection Pressure Strain Stress Temperature

**NOTE:** If data processing mode is turned on, new documents will be opened in data processing

### ✓JR √Viewer

### XJR XViewer

✓JR √Viewer

### © 2014 HydeSoft Computing, LLC

### ✓JR Xiewer

mode. This may not always be appropriate. Applications for DPlot written by other developers should always turn data processing mode **off** after closing all documents.

## 5.2.57 DataProcessLabels command

### [DataProcessLabels(state)]

Equivalent to checking (*state*=1) or unchecking (*state*=0) the Use DPlot-generated axis labels box on the Data Processing dialog.

### 5.2.58 DateFormat command

### [DateFormat("format")]

The **format** string controls the appearance of numbers using the *Calendar date* or *Date and time* <u>number formats</u>. This is the same date format string that will appear in the *Date Format* box of the <u>Number Formats</u> dialog.

✓JR √Viewer

To display	Use
Numbers with no leading zeros for single-digit days, years, or months.	Lowercase <b>d</b> or <b>y</b> , or uppercase <b>M</b>
Numbers with leading zeros for single-digit days, years, or months.	Lowercase <b>dd</b> or <b>yy</b> , or uppercase <b>MM</b>
Three-letter abbreviations for the day of the week or month.	Lowercase <b>ddd</b> or uppercase <b>MMM</b>
The full name of the day of the week, year, or month	Lowercase <b>dddd</b> or <b>yyyy</b> , or uppercase <b>MMMM</b>
The day of the year (Jan 1=1, Dec 31=365 or 366)	Lowercase <b>n</b>

### 5.2.59 DecimalDigits command

### 

Sets the number of decimal places used to display numbers when the number format is set to *Default* or *Percent* for the X axis (*axis*=0), Y axis (*axis*=1), Z axis or secondary X axis on XY plots( *axis*=2), or secondary Y axis on XY plots (*axis*=3). If *digits* = -1, DPlot uses its default formatting: no trailing zeros, maximum of seven significant figures. This setting is ignored for amplitudes greater than or equal to 10,000,000 or less than 0.00001.

Also see NumberFormat.

### 5.2.60 DeleteLabel command

### [DeleteLabel(index)]

✓JR XViewer

Deletes a label created with the <u>TextPointLabel</u> command or <u>Label Points</u> menu command. Deletes the *index*'th label (1-based). If *index* is less than 0, all labels are deleted.

### XJR XViewer

## 5.2.61 DeferMinMaxCheck command

### [DeferMinMaxCheck(state)] ✓JR XViewer

This command is only useful when calling the DPlot AddData DPLOTLIB function or when using the XY, XYXY, or XYZ commands from a different program. If state is non-zero, then the minimum values, maximum values, mean, standard deviation and several other parameters are not recalculated after every call to theses functions/commands. Those calculations tend to take a long time (long as in more than a few milliseconds) when a plot contains many thousands of points. This bottleneck could be the difference between success and failure in real-time applications with a short time step (less than 0.1 seconds). This is particularly true when the calling application is written with VB.NET, which for some reason is a bit sluggish with DDE calls. [DeferMinMaxCheck(0)] turns this setting off and immediately causes DPlot to calculate those values and redraw the plot. The downside to using [DeferMinMaxCheck(1)] is that text placeholders will not be updated after every call to DPlot AddData or when using the XY, XYXY, or XYZ commands.

The extents and other values will be recalculated and a plot be redrawn with DeferMinMaxCheck(0) if that state was previously non-zero.

## 5.2.62 Deletelmage command

### [Deletelmage(index)]

Deletes an image added with the InsertImage macro command or the Insert Image command on the Options menu. If *index* is omitted, this command deletes the last image added.

✓JR XViewer

See also

Insert Image menu command InsertImage macro command

### 5.2.63 DeleteNote command

### [DeleteNote(index)]

Deletes a notation created with the TextNote, TextNote3D, TextNoteEx commands or Add/Edit Note menu command. Deletes the index the note (1-based). If index is less than 0, all notes are deleted. If index is 0, the last-created note is deleted.

### See also

Add/Edit Note Text menu command

### 5.2.64 DeleteRefLine command

### [DeleteRefLine(index)]

Deletes a reference line created with the RefLine command or Reference Lines menu command. Deletes the *index* th line (1-based). If *index* is less than 0, all lines are deleted. If *index* is 0, the lastcreated reference line is deleted.

✓JR XViewer

### See also

Reference Lines Options menu command

### © 2014 HydeSoft Computing, LLC

### ✓JR XViewer

## 5.2.65 DeleteThinTriangles command

### [DeleteThinTriangles(angle)]

Removes triangles with interior angles less than *angle* if the associated triangle vertex is on a free edge. Thin triangles often do a poor job of mapping a surface. Triangle removal is recursive: if triangle B has an interior angle less than *angle* which borders triangle A, and triangle A is removed, then triangle B will also be removed.

### 5.2.66 DenseGridLines command

### [DenseGridLines(nx/,ny/)]

Sets the number of divisions for minor grid lines on the X and Y axes (or angular and radial axes for polar plots). A 0 value results in no minor grid lines/tick marks. If **ny** is omitted it is set equal to the **nx** value. If **ny** is omitted and **nx**=1, both **nx** and **ny** are set to 4 for compatibility with earlier versions of this command that only turned the feature on or off.

### 5.2.67 Differentiate command

### [Differentiate(|method,replace/)]

Differentiates (finds  $\partial y/\partial x$ ) the curve or curves selected with the <u>SelectCurve</u> command. Both *method* and *replace* are optional. Use *method*=1 for the *Central difference* method, =2 for *Quadratic interpolation*, or =3 for the *Previous point* method. If *method*=2 and the source curve does **not** have evenly-spaced points, *method*=1 will be used instead. If *method* is omitted, then *Quadratic interpolation* is used for evenly-spaced points and *Central difference* is used otherwise.

Set *replace* to a non-zero value to replace the source curve with the result of this function, or 0 to create a new curve. Default (omitted) = 0.

### See also

Differentiate a Curve menu command

### 5.2.68 Directory command

### Directory("*dir*")] or Directory(*code*)

Sets the directory for all subsequent <u>FileOpen</u> and <u>FileSaveAs</u> operations. If the filenames used in a FileOpen or FileSaveAs command begin with a drive letter or backslash (\), the Directory entry is ignored. You can specify a relative path in FileOpen and FileSaveAs commands by including the desired relative path, but **not** a leading backslash. For example:

✓JR √Viewer

Directory("c:\my data")
FileSaveAs(1,"test123\gage001.grf")

will save the current plot to the file c:\my data\test123\gage001.grf, assuming the test123 subdirectory exists.

Valid code's are:

0 Set to My Documents\DPlot

1 Set to the last folder that was used in a manual <u>File>Open</u> or <u>File>Append</u> menu command.

2 Set to the last folder that was used in a manual <u>File>Save As</u> menu command.

If the "dir" entry is two consecutive quotation marks, as in Directory(""), or completely omitted ( Directory()) the default directory is set to My Documents\DPlot.

Please note: This setting persists only during the current DDE conversation, and will be reset to the

XJR XViewer

# ✓JR ×Viewer

✓JR XViewer

previous default folder on subsequent conversations. All commands in a single macro are executed during the same conversation. But <u>DPLOTLIB</u>'s DPlot_Command function starts a new conversation on each call. So this:

```
DPlot_Command("[Directory(\"c:\\my data\")]");
DPlot_Command("[FileSaveAs(1,\"test123\\gage001.grf\")]");
```

will save the file to <default folder>\test123\gage001.grf.

The *dir* parameter may include the shortcuts **[DPLOTDRIVE]**, **[DPLOTPATH]**, or **[DPLOTDOCS]** to set the working directory to the drive or full path where DPlot is installed or, for [DPLOTDOCS], to My Documents\DPlot. All forms must be uppercase and be surrounded by the pipe symbol (|) or will be ignored. Please note that the **[DPLOTPATH]** destination folder may not be acceptable for output with non-admin accounts on Windows XP and all accounts on Windows Vista and newer versions, depending on where DPlot is installed.

For example:

Directory("|DPLOTPATH|")

sets the working directory to the path where dplot.exe (or dplotjr.exe or dplotvu.exe) are installed.

The trailing colon will be included with **|DPLOTDRIVE|**; similarly the trailing backslash will be included with **|DPLOTPATH|** and **|DPLOTDOCS|**. The drive letter and colon are included in **| DPLOTPATH|**; there is no need to use, for example, "**|DPLOTDRIVE||DPLOTPATH|**".

### 5.2.69 DivideCurves command

### [DivideCurves(*Curve1,Curve2*)] XJR Viewer

Generates a new curve with amplitudes equal to the amplitudes of *Curve1* divided by the amplitudes of *Curve 2*.

X values for the new curve are taken from *Curve1*, subject to whether <u>FXY1Y2_XFrom</u> with a nonzero *state* value has been previously used. Linear interpolation is used to find the amplitude of *Curve2* at each X.

See also <u>Y=f(X,Y1,Y2,...)</u> menu command

### 5.2.70 DocHide command

[DocHide()]

### ✓JR √Viewer

✓JR √Viewer

Hides the document window referenced by the DDE conversation or macro.

### 5.2.71 DocMaximize command

### [DocMaximize()]

Maximizes and activates the document window referenced by the DDE conversation or macro.

### 5.2.72 DocMinimize command

### 

Minimizes (reduces to an icon) the document window referenced by the DDE conversation or macro.

### 5.2.73 DocRestore command

### [DocRestore()]

Restores (to unmaximized or unminimized size) the document window referenced by the DDE conversation or macro.

### 5.2.74 DocShow command

### [DocShow()]

Shows (after hiding) the document window referenced by the DDE conversation or macro.

### 5.2.75 EditAddToX command

[EditAddToX(value)]

Adds **value** to X. Operates on the curve or curves selected with the SelectCurve command.

### 5.2.76 EditAddToY command

Viewer
V

Adds *value* to Y. Operates on the curve or curves selected with the SelectCurve command.

### 5.2.77 EditDecimate command

### [EditDecimate(*method*,*replace*,*value*],*start*, XJR XViewer stop/)]

Decimates data values in the curve or curves selected with the <u>SelectCurve</u> command. *Method* = 0 to average every **value** points, = 1 to discard all but every **value**'th point, or = 2 to use the Welch depopulation scheme with fidelity = value. Replace = 1 to replace existing data points, = 0 to preserve the original data and produce a new curve. Operates on data points between X = start and X = stop. Start and stop are each optional; if start is omitted then the minimum X value for the curve is used; if stop is omitted then the maximum X value for the curve is used. If both are omitted then the entire curve will be operated on.

### 5.2.78 EditEqualInts command

[EditEqualInts(replace.interval.method]. XJR Viewer specify limits,startX,endX/)]

Generates a new curve with a constant spacing of *interval* on the abscissa (X axis), using linear ( *method* = 1), quadratic (*method* = 2), cubic (*method* = 3), or cubic spline (*method* = 4) interpolation. Replaces the existing curve if *replace* is not equal to 0, otherwise generates a new curve. If the optional specify limits is 0 or not specified, then the minimum and maximum X values for each selected curve will be used in the new curve(s). If *specify_limits* is non-zero, then the new curve(s) will start at X=startX and end at X=endX.

Operates on the curve or curves selected with the SelectCurve command. For more information see the description of the Equal Intervals command.

### 5.2.79 EditErase command

### [EditErase()]

✓JR XViewer

XJR XViewer

✓JR √Viewer

✓JR ✓Viewer

Deletes the curve most recently selected with the <u>SelectCurve</u> command. After the succesful completion of this command the curve selection becomes undefined.

### 5.2.80 EditEraseData command

### 

Similar to EditErase, but rather than deleting a curve it simply sets the number of points for the given *curve* to 0. This command also does not redraw the plot, delete the legend or other text labels associated with the given *curve*, force the plot extents to be recalculated, or close the document if this operation results in deleting all existing data. This command is primarily useful in realtime applications that plot, for example, the last X minutes of data. If *curve* is less than 0, this operation sets the number of points for **all** curves to 0.

### 5.2.81 EditFilter command

[EditFilter("user",passes,replace,C0,C1,C2, XJR XViewer D1,D2)] or [EditFilter("bandpass",passes,replace,freq1, freq2|,timeunits|,stop_or_pass|/)] or [EditFilter("optionX",passes,replace)] or [EditFilter("cfcX",passes,replace)]

Applies a digital filter to the curve most recently selected with the <u>SelectCurve</u> command.

"USER" filter: **C0**, **C1**, **C2**, **D1**, and **D2** are the coefficients of the filter equation to be used (described in the <u>Filter</u> topic).

"BANDPASS" filter: If *stop_or_pass* is 0 or omitted and *freq1* is a positive number, then *freq1* is the frequency to remove from the data record and *freq2* is the bandwidth, or width of the notch, to remove. If *freq2* is equal to 0 then the Carleton method is used; h otherwise the Press method is used. If *freq1* is equal to 0, the zero frequency rejection filter is used and *freq2*, *timeunits*, and *stop_or_pass* are ignored. If *stop_or_pass* is non-zero, then *freq1* specifies the lower cutoff frequency and *freq2* specifies the upper cutoff frequency. To implement a low-pass filter, set *freq1* to 0; to implement a high-pass filter set *freq2* to a very high frequency.

*timeunits* indicates the units of time for the input X values. If 0 or omitted, seconds are assumed. Set to 1 for milliseconds, 2 for microseconds.

"OPTIONX" filter: X must be one of 1, 2, 3, or 4 and corresponds to the filters described in the <u>Filter</u> topic.

"CFCX" filter: X must be one of 60, 180, 600, or 1000 and corresponds to the filters described in the <u>Filter</u> topic. Time units are determined by the X axis label. If the label includes the characters "msec" or "millisec", milliseconds are assumed. If the label includes "microsec" then microseconds are assumed. Otherwise seconds are assumed.

**Passes** is a misnomer for historical reasons and is **one less** than the number of passes through the data. The operation is performed in forward, reverse, forward, reverse, etc. order. So if **passes** is 0, 2, 4, etc. the operation will end in a forward pass. If 1, 3, 5, etc. the operation ends with a time-reversed pass.

If *replace* is non-zero, then the original data set is replaced by the filtered data. If *replace* is equal to 0, then the original data is preserved.

### See also

Filter menu command

## 5.2.82 EditMultiplyX command

[EditMultiplyX(value)] XJR XViewer Multiplies X by value. Operates on the curve or curves selected with the SelectCurve command.

## 5.2.83 EditMultiplyY command

[EditMultiplyY(value)] XJR XViewer Multiplies Y by value. Operates on the curve or curves selected with the SelectCurve command.

### 5.2.84 EditNormalize command

### [EditNormalize(option)]

Adjusts the amplitudes of the curve or curves selected with the SelectCurve command, so that they range from 0-1 (*option*=0) or 0-100 (*option*=1).

### See also

Normalize menu command

### 5.2.85 EditOperateX command

### [EditOperateX("equation")]

For XY plots, modifies the X values of the curve or curves selected with the SelectCurve command by transforming them with *equation*, where *equation* is a function of the existing X and/or Y values. For example, if equation="X*2", then all X values are shifted to the right by 2.

XJR XViewer

See also

Operate on X menu command

### 5.2.86 EditOperateY command

### [EditOperateY("equation")]

For XY plots, modifies the Y values of the curve or curves selected with the SelectCurve command by transforming them with *equation*, where *equation* is a function of the existing X and/or Y values. For example, if equation="Y^2", then all Y values will be multiplied by themselves.

### See also

Operate on Y menu command

### 5.2.87 EditOperateZ command

### [EditOperateZ("equation")]

For 3D data, modifies the Z values of the curve or curves selected with the SelectCurve command by transforming them with *equation*, where *equation* is a function of the existing X, Y, and/or Z values. For example, if *equation*="Z*1000", then all Z values are multiplied by 1000.

×JR ×Viewer

### See also

Operate on Z menu command

© 2014 HydeSoft Computing, LLC

XJR XViewer

## XJR XViewer

### 5.2.88 EditRemoveSpikes command

### [EditRemoveSpikes(*SpikeLow,SpikeHigh,* XJR XViewer *SpikeRatio,X1,X2*)]

Removes single samples with high amplitudes which bear no relation to the true record from the curve or curves specified with the <u>SelectCurve</u> command. This command performs the same operation as the Remove Random Spikes command on the Edit menu.

The conditions defining a spike at point n are:

SpikeLow 
$$< \frac{y_n - y_{n-1}}{y_n - y_{n+1}} <$$
SpikeHigh

The condition which defines the sizes of spikes to be removed is:

$$\left|y_{n} - \left(\frac{y_{n-1} + y_{n+1}}{2}\right)\right| > SpikeRatio \times Range$$

where Range is the current range in amplitudes for the record. If any sample in a record meets both of the conditions described above, it is considered a spike and replaced with the average of the adjacent values.

All parameters are optional. If *SpikeLow*, *SpikeHigh*, or *SpikeRatio* are omitted, default values of 0.75, 1.25, and/or 0.1, respectively, will be used. All points in a curve will be operated on unless both *X1* and *X2* (lower and upper limits) are specified.

### See also

Remove Random Spikes menu command

### 5.2.89 EditRemoveTrend command

### [EditRemoveTrend()]

Removes any linear trend from the curve or curves selected with the <u>SelectCurve</u> command, so that the first and last amplitudes are equal to 0.

See also

Remove Trend menu command

### 5.2.90 EditReplaceData command

### [EditReplaceData(destination,source)]

Replaces the data points in an XY plot for curve with 1-based index **destination** with the data from curve with index **source** and deletes curve **source**. This command is particularly useful when you need to update an existing saved plot with updated data. Legend, point labels, symbol and line styles, and drawing order for the **destination** curve are preserved.

XJR XViewer

Example:

[EditReplaceData(2,5)]

replaces the data points in curve 2 with the data from curve 5, and deletes curve 5.

[EditReplaceData(3,LegendHas("New Curve")] replaces the data points in curve 3 with data from the first curve (if any) whose legend contains

### XJR XViewer

the character string "New Curve", then deletes that curve.

This command results in an error if either *destination* or *source* are less than 1 or greater than the number of curves in the active plot, or if *destination=source*.

Numeric curve indices may be replaced by substituting LegendHas

("substring1"],"substring2"],"substring3"]]). When this feature is used, DPlot will search for a curve whose legend entry contains substring1 and optionally substring2 and substring3. The index of the first curve whose legend matches that criteria will be substituted for LegendHas(...). If no match is found, no substitution is performed. Searches are case-specific.

For example, [SelectCurve(LegendHas("ABC", "GHI"))] will search for a curve whose legend contains both ABC and GHI.

"ABCDEFGHI" is a positive match; "ABCDEFGH" is not. Neither is "abcdefghi".

LegendHas(...) may be used in any command in place of a curve or data set index.

### 5.2.91 EditSmooth command

### [EditSmooth(*replace,window*],*X1,X2*])] XJR Viewer

Smooths the amplitudes of the curve or curves selected with the <u>SelectCurve</u> command. The smoothing **window** is the number of points over which the data should be smoothed. Zero gives no smoothing at all, while a value approaching half the number of data points will render the data virtually featureless. The existing curve will be replaced by the new curve if **replace** = 1, otherwise the existing curve will be preserved. If **X1** and **X2** are specified, this command only operates on points between those extremes; points outside that range are unaffected. If **X1** and **X2** are omitted, the command operates on all points. For more information see the Help topic on the <u>Smooth</u> command.

### 5.2.92 EditSort command

### [EditSort()]

Sorts the data for the curve or curves selected with the <u>SelectCurve</u> command in ascending abscissa (X) order, then in ascending Y order for equal X values. This macro command is equivalent to selecting the Sort command on the Edit menu.

### 5.2.93 EditSubtractMean command

### [EditSubtractMean()]

Subtracts the average amplitude (Y value) from all Y values of the curve or curves selected with the <u>SelectCurve</u> command.

### 5.2.94 EditSwapXY command

### [EditSwapXY()]

Exchanges the X and Y values for the curve or curves selected with the <u>SelectCurve</u> command.

### 5.2.95 EditTruncate command

[EditTruncate(from, to)]

XJR XViewer

### XJR XViewer

# XJR XViewer

×JR ×Viewer

Eliminates all data outside the range specified by  $from \le X \le to$ . Operates on the curve or curves selected with the <u>SelectCurve</u> command.

### 5.2.96 Enable command

### [Enable(state)]

Enables (*state* <> 0) or disables (*state* = 0) DPlot user interface. Although DPlot temporarily disables toolbar buttons and menu commands during DDE transactions, it is still possible for a user to choose a menu command between transactions. If the transactions are being sent to DPlot at a fairly fast rate (e.g. during real-time data processing), it is possible that DPlot will crash in response to the user selecting a menu command that causes a modal dialog box to open. To prevent this situation, you can prevent the user from selecting a menu command by sending "[Enable(0)]". DPlot will re-enable itself after the DDE conversation, or link, has been terminated.

✓JR √Viewer

### 5.2.97 ErrorBar command

[ErrorBar(*curve,display,type,param1,* **√**JR **√**Viewer *param2*/,*color,frequency*/)]

Sets error bar parameters for curve number *curve*. Set *display*=0 to turn error bars off, 1 to display positive error bars, 2 to display negative error bars, or 3 to display both positive and negative error bars. Add 4 for horizontal bars. If *type*=0, *param1* specifies a fixed error bar value and *param2* is ignored. If *type*=1, *param1* specifies a percentage value for the error bars; *param2* is ignored. If *type*=2, *param1* and *param2* specify absolute positive and negative error values, respectively. If type=3, *param1* and *param2* specify curve indices (1-based) to retrieve positive and negative error values from. If used, those curves are used **only** as error values and are not drawn.

*color* and *frequency* are optional parameters. *color* specifies the color of the error bars, = blue*256²+green*256+red. The color may also be expressed in hexadecimal as 0x00bbggrr. The *frequency* setting controls how many error bars are drawn. If *frequency*=0 or 1 (or is omitted), error bars are drawn at every data point, if=2, error bars are drawn at every other point, etc.

### See also

Error Bars menu command ErrorBarPenWidth macro command ErrorBarWidth macro command

### 5.2.98 ErrorBarPenWidth command

### [ErrorBarPenWidth(width)]

✓JR √Viewer

Specifies the width of the pen used to draw error bars, in thousandths of inches. Default value=20.

See also

Error Bars menu command ErrorBar macro command ErrorBarWidth macro command

### 5.2.99 ErrorBarWidth command

### [ErrorBarWidth(state)]

✓JR ✓Viewer

Specifies the width of error bars, in thousandths of inches. Default value=250.

See also Error Bars menu command ErrorBar macro command ErrorBarPenWidth macro command

## 5.2.100 ErrorMessages command

### ✓JR √Viewer [ErrorMessages(state)]

Turns error messages for DDE errors encountered by DPlot on (state <> 0) or off (state = 0). The error message state is turned on (non-zero) whenever a macro or DDE conversation is initiated. For applications that process their own error messages by interpreting the results of DdeClientTransaction, it may be desirable to turn error messages off.

## 5.2.101 FFT command

[FFT()]

Performs a Fast Fourier Transform on the curve selected with the SelectCurve command. The FFT command creates a new document window for displaying the output. For more information, see the Help topic for the FFT menu command.

### 5.2.102 FFTMethod command

### [FFTMethod(method)]

If *method* is equal to 0, DPlot pads the input record with zeroes out to the next highest power of 2 so that a more efficient FFT algorithm can be used. If method is not equal to 0, DPlot does not modify the input record. Using method  $\neq 0$  may result in very long run times, depending on the number of points in the input record. This command takes effect with the next FFT command. For more information, see the Help topic for the FFT menu command.

### 5.2.103 FFTPhase command

[FFTPhase(state)]

If state  $\neq 0$ , DPlot generates a phase angle plot along with the FFT amplitudes. This command takes effect with the next FFT command.

## 5.2.104 FileAppend command

[FileAppend("filename")] XJR √Viewer

or [FileAppend("filename","option,option,

option,...") (file import plugins only)

> Opens a file, creating one or more new curves in the currently active document. The file type must not be a DPlot file. The FileType command should be issued before the FileAppend command.

> The *filename* parameter may include the shortcuts **[DPLOTDRIVE]** or **[DPLOTPATH]** to include the drive or full path, respectively, where DPlot is installed. This is most often useful when DPlot is installed to a removable drive. Either form must be uppercase or will be ignored. For example:

FileAppend("|DPLOTDRIVE|\datafiles\myfile.csv")

XJR XViewer

# XJR XViewer

XJR XViewer

will attempt to open the file myfile.csv located in the \datafiles folder on the drive where DPlot is installed.

The trailing colon will be included with **|DPLOTDRIVE|**; similarly the trailing backslash will be included with **|DPLOTPATH|**. The drive letter and colon are included in **|DPLOTPATH|**; there is no need to use, for example, "**|DPLOTDRIVE||DPLOTPATH|***filename*".

The second form of this command is only useful for file import plugins that include the function <u>plugin readfileex</u>. The interpretation of the options string is entirely up to the plugin and is passed unchanged to the plugin_readfileex function. Generally this options string is used to accept options for the plugin programmatically rather than forcing user input via a dialog box. Options for file import plugins distributed with DPlot are described below.

### **AMO Transient Recorder Files**

 Options are comma-separated values.

 Option, Start_Time, Stop_Time

 where:

 Option
 0=read entire record.

 1=read 1st preview block.

 2=read 2nd preview block.

 3=read 3rd preview block.

 4=read full record between Start_Time and Stop_Time (seconds).

 Start_Time and Stop_Time are optional and are ignored for all but Option=4.

### **ArcView Shapefiles**

Only one option available: Separate where:

Separate

If non-zero, different entities within the file will be treated as separate curves. If 0, different entities will be combined into the same curve, with a marker point with latitude = 100 degrees inserted between entities. For all shapefiles, the plugin turns on <u>Mercator Projection</u> scaling, which in turn forces <u>Amplitude Limits</u> to +/-85 degrees. For *Separate* = 0, this will in effect split the shapefile entities into separate pieces, though they will remain part of the same data set.

If a shapefile contains more entities than the maximum number of curves (100 minus the number of existing curves), this option is ignored and all entities will be combined.

### 5.2.105 FileArrays command

### [FileArrays(maxc,maxp)]

✓JR √Viewer

Sets the maximum number of curves and the maximum number of points per curve. DPlot will **not** plot more points than *maxp* per curve or create more curves than *maxc*.

### 5.2.106 FileClose command

### [FileClose()]

✓JR ✓Viewer

Closes the document referenced by the current DDE conversation or macro. It is not possible to predict which document will be associated with the DDE conversation or macro after a **FileClose** command. If any subsequent commands are to be executed, they should be preceded by an **Activate**, <u>FileNew</u>, or <u>FileOpen</u> command.

## 5.2.107 FileExit command

### [FileExit()]

✓JR √Viewer

Closes DPlot. **Developers should use care in issuing this command**, since it is possible that the user may have opened DPlot for another purpose.

### 5.2.108 FileMacro command

[FileMacro()]

Displays the macro dialog box.

### 5.2.109 FileNew command

[FileNew()]

✓JR √Viewer

XJR √Viewer

Finds an empty document window. If no empty document window exists, **FileNew** opens a new document window and sets the plot type to X,Y plot. Subsequent commands in the same DDE conversation or macro reference the newly opened document.

[FileNew(1)]

Identical to [FileNew()], but sets the plot type to 1D statistics plot.

[FileNew(3)]

Identical to [FileNew()], but sets the plot type to 3D random points (contour plot).

### 5.2.110 FileNewEx command

### 

Finds an empty document window or opens a new document window if no empty documents exist. The caption of the new document is set to "*caption*". Plot type is set to 1D statistics plot for *datatype* =1, 2D XY plot for *datatype*=2, or 3D random points for *datatype*=3. If *activate* is non-zero, the new document is activated and becomes the target of all subsequent DDE commands. If *activate*=0, the currently active document remains the active document.

## 5.2.111 FileOpen command

[FileOpen("*filename*")] ×JR √Viewer or [FileOpen("*filename*","*option,option,option,...* ")

### (file import plugins only)

Opens and displays a file in a new or empty document window. If the file type is **not** a DPlot file, then the <u>FileType</u> command should be issued **before** the **FileOpen** command. Subsequent commands reference the document created by this command. The path to the file should either be absolute (as in "c:\datafiles\mydata.dat") or relative to the current directory. By default, the current directory is the My Documents\DPlot folder. The current directory can be changed with a <u>Directory</u> command. You can embed Windows' environment variables in filenames, delimited with percent (%) symbols. For example "%USERPROFILE%\My Documents\DPlot\test.csv" will be interpreted as "c: \Documents and Settings\cusername>\My Documents\DPlot\test.csv" on Windows Vista and newer versions of Windows.

The *filename* argument may include the shortcuts **[DPLOTDRIVE]**, **[DPLOTPATH]**, or **[** 

**DPLOTDOCS** to set the input drive or full path, respectively, to where DPlot is installed or, for | **DPLOTDOCS**, to My Documents\DPlot. This is most often useful when DPlot is installed to a removable drive. All forms must be uppercase and surrounded by the pipe symbol (|) or will be ignored. For example:

FileOpen("|DPLOTDRIVE|\datafiles\myfile.grf")

will attempt to open the file myfile.grf located in the \datafiles folder on the drive where DPlot is installed.

The trailing colon will be included with **|DPLOTDRIVE|**; similarly the trailing backslash will be included with **|DPLOTPATH|**. The drive letter and colon are included in **|DPLOTPATH|**; there is no need to use, for example, "**|DPLOTDRIVE||DPLOTPATH|***filename*".

The second form of this command is only useful for file import plugins that include the function <u>plugin readfileex</u>. The interpretation of the options string is entirely up to the plugin and is passed unchanged to the plugin_readfileex function. Generally this options string is used to accept options for the plugin programmatically rather than forcing user input via a dialog box. Options for file import plugins distributed with DPlot are described below.

### **AMO Transient Recorder Files**

Options are comma-separated values.

*Option, Start_Time, Stop_Time* where:

Option

n 0=read entire record. 1=read 1st preview block.

2=read 2nd preview block.

3=read 3rd preview block.

4=read full record between Start Time and Stop Time (seconds).

Start Time and Stop Time are optional and are ignored for all but Option=4.

### **ArcView Shapefiles**

Only one option available: Separate where:

 Separate
 If non-zero, different entities within the file will be treated as separate curves.

 If 0, different entities will be combined into the same curve, with a marker point with latitude = 100 degrees inserted between entities. For all shapefiles, the plugin turns on Mercator Projection scaling, which in turn forces

 Amplitude Limits to +/-85 degrees. For Separate = 0, this will in effect split the shapefile entities into separate pieces, though they will remain part of the same data set.

 If a shapefile contains more entities than the maximum number of curves

(100 minus the number of existing curves), this option is ignored and all entities will be combined.

### Bitmap to 3D

Options are comma-separated values. All entries are optional and are read in the order *Scale*, *Black_Value*, *White_Value*, *Filter*, *Left*, *Top*, *Right*, *Bottom*, *Xlo*, *Ylo*, *Xhi*, *Yhi* where:

Scale	Integer from 10 to 100. Controls how many columns and rows the surface plot will have. At 100, the number of data columns will be equal to the width of the selected part of the image; the number of data rows will be equal to the height. Default=100.
Black_Value	Z value mapped to color index 0 in the (possibly converted) 8-bit gray scale image. Default=0.
White_Value Filter	Z value mapped to color index 255 in the image. Default=255. For scales less than 100%, specifies the resampling filter to use on the

image. 0=bilinear, 1=B-spline, 2=bicubic. Default=2.

Left, Top, Right, BottomSpecifies the 0-based extents of the portion of the image to use, in<br/>pixels. 0,0 is the top left corner of the image. Default is to use entire image.XIo, YIo, Xhi, YhiMaps Left, Bottom, Right, Top image pixels to X and Y values in the generated<br/>surface plot. Default is 0,0 to width-1, height-1.

All entries are optional, but you must include all previous options. For example you cannot specify *Filter* without also specifying *Scale*, *Black_Value*, and *White_Value*. If at least one option is specified then the default options shown above are used for any omitted options.

### **Pacific Instruments EU Files**

Options are comma <i>Start_Channel,Stop</i> where:	a-separated values. All entries are optional and are read in the order p_Channel,Sample_Count,UseFireSwitch,TimeStartsAt0
Start_Channel	0-based index of the first channel to read. Ignored for <i>Stop_Channel</i> < 0. Default=0.
Stop_Channel	0-based index of the last channel to read. For values less than 0, <b>all</b> channels will be included. Default=-1.
Sample_Count	Number of samples to read, starting from the start of the file. Use -1 to read all samples. Default=-1.
UseFireSwitch	Non-zero to use fire switch time - first time = -fire switch time. If 0, first time is retrieved from the file header. Default=0.
TimeStartsAt0	If non-zero <b>and</b> UseFireSwitch = 0, first time = 0. Default=0.

All entries are optional, but you must include all previous options. For example you cannot specify *Sample_Count* without also specifying *Start_Channel* and *Stop_Channel*. If at least one option is specified then the default options shown above are used for any omitted options.

### 5.2.112 FilePageSetup command

### [FilePageSetup(top,bottom,left,right, orientation,papersize,width,height)]

✓JR √Viewer

Sets the printer margins (in inches) for the currently selected document, and optionally sets the page orientation and paper size. **All** parameters are optional and **must** be separated by commas. Omitted parameters are represented by 2 successive commas with no intervening spaces. If **orientation** is included, subsequent printouts will default to Landscape orientation for **orientation=**2, or Portrait orientation otherwise. **papersize** must be included (and set to 0) if you want to specify the **width** and **height**, in inches. Valid codes for **papersize** are:

0	User-defined size	22	Envelope #12 4.75 x 11 in
1	Letter 8.5 x 11 in	23	Envelope #14 5 x 11.5 in
3	Tabloid 11 x 17 in	24	C size sheet
4	Ledger 17 x 11 in	25	D size sheet
5	Legal 8.5 x 14 in	26	E size sheet
6	Statement 5.5 x 8.5 in	27	Envelope DL 110 x 220 mm
7	Executive 7.25 x 10.5 in	28	Envelope C5 162 x 229 mm
8	A3 297 x 420 mm	29	Envelope C3 324 x 458 mm
9	A4 210 x 297 mm	30	Envelope C4 229 x 324 mm
10	A4 Small 210 x 297 mm	31	Envelope C6 114 x 162 mm
11	A5 148 x 210 mm	32	Envelope C65 114 x 229 mm

12	B4 (JIS) 250 x 354 mm	33	Envelope B4 250 x 353 mm
13	B5 (JIS) 182 x 257 mm	34	Envelope B5 176 x 250 mm
14	Folio 8.5 x 13 in	35	Envelope B6 176 x 125 mm
15	Quarto 215 x 275 mm	36	Envelope 110 x 230 mm
16	10x14 in	37	Envelope Monarch 3.875 x 7.5 in
17	11x17 in	38	6 3/4 Envelope 3.625 x 6.5 in
18	Note 8.5 x 11 in	38	US Std Fanfold 14.875 x 11 in
19	Envelope #9 3.875 x 8.875 in	40	German Std Fanfold 8.5 x 12 in
20	Envelope #10 4.125 x 9.5 in	41	German Legal Fanfold 8.5 x 13 in
21	Envelope #11 4.5 x 10.375 in		

**Please note:** Many printer drivers will not accept all of the paper size codes above, and most will not accept custom width and height values unless they happen to match dimensions supported by that printer. If a paper size is rejected by a selected printer, the default paper size for that printer will be selected. Virtual printer drivers like Acrobat Distiller will generally accept any custom paper size.

### 5.2.113 FilePrint command

### 

Prints the document using the current default printer settings.

### [FilePrint(1)]

Displays the Print dialog box, from which the user may change printer settings and print the currently active document and optionally print any or all of the other open documents on the same page.

### [FilePrint("printer","filename")]

Prints the document using the default driver and port settings for the *printer*, where *printer* is the printer description **exactly** as it appears in the Control Panel Printers applet and/or the Print dialog in DPlot. *Printer* must be installed on this system. It is not restricted to physical printers; it may be a virtual printer like Acrobat Distiller or Adobe PDF, for example. If the optional *filename* parameter is included and print output is to a file (rather than an actual printer), then the output will be saved to this name. If *filename* is omitted and output is to a file then generally the plot title with the appropriate extension (e.g. .PDF, .EPS, .PRN) will be used and the file will be saved to the user's My Documents folder. (In general, though, this decision is ultimately up to the selected printer driver if you do not specify a filename.)

The *filename* argument may include the shortcuts **|DPLOTDRIVE|**, **|DPLOTPATH|**, or **| DPLOTDOCS|** to set the destination drive or full path, respectively, to where DPlot is installed or, for **|DPLOTDOCS|**, to My Documents\DPlot. This is most often useful when DPlot is installed to a removable drive. All forms must be uppercase and surrounded by the pipe symbol (|) or will be ignored. Please note that the **|DPLOTPATH|** destination folder may not be acceptable to non-admin accounts on Windows XP and all accounts on Windows Vista and newer versions, depending on where DPlot is installed.

**Please note:** All PDF printer drivers tested with this command work as expected if no *filename* is specified: the user is prompted for a filename and gets the expected PDF format. If a *filename* is specified, this is handled by various drivers in different ways, most of them unexpected. Without the workarounds inserted in DPlot and described below, some drivers will produce a PostScript file rather than PDF format (notably the Adobe PDF driver), while others insist on prompting for a filename whether one is specified or not. One exception is Win2PDF, from <u>Dane Prairie Systems</u>, which works as expected when a *filename* is specified: The Save As dialog is not displayed, and the output is PDF.

DPlot includes code to work around this behavior for print drivers Adobe PDF, Pdf995 from <u>Software995</u>, PDFcamp Printer from <u>veryPDF</u>, and docuPrinter from <u>Neevia Technology</u>. Unfortunately the <u>CutePDF</u> driver is a problem: With no workarounds and a filename specified in the FilePrint command, CutePDF produces a PostScript file. With workarounds specified by CutePDF, the Save As dialog **is** displayed even though the workaround specifically tells the driver not to display it. Current versions of DPlot will write to a PDF file using CutePDF, but the Save As dialog will be displayed whether a filename is specified or not.At this writing, CutePDF is not recommended for use in a FilePrint command for other than your own internal use.

For programmers interested in this sort of thing:

Adobe PDF: Modify registry key HKEY_CURRENT_USER\Software\Adobe\Acrobat Distiller\PrinterJobControl to include:

<full path to dplot. filename exe> or <windir>\splwow64. exe on 64-bit Windows versions

This registry entry is automatically deleted by the Adobe print driver once StartDoc is called.

Pdf995: Set lpszDocName member of DOCINFO structure to <pdf995: filename>

**PDFcamp:** Modify registry key HKEY_CURRENT_USER\Software\verypdf\pdfcamp to include the following values:

AutomaticDir*filename* ectory AutomaticOu1 (enables AutomaticDirectory) tput

AutomaticVa2 (overwrite if file exists) lue

AutoView 0 (does not open PDF file after creation)

These values are restored to their previous settings (or deleted if they did not previously exist) once the file is created. For more information on PDFcamp registry settings see <u>How do I make an</u> <u>output file without bringing up the "Save As" dialog?</u> on the veryPDF web site.

**docuPrinter:** Modify registry key HKEY_CURRENT_USER\Software\NEEVIA\Neevia docuprinter to include the following values:

Filemask **filename** w/o path

OutputPath path of *filename* 

Option_Save 0

As_Visible

These values are restored to their previous settings (or deleted if they did not previously exist) once the file is created.

### 5.2.114 FilePrintMultiple command

[FilePrintMultiple( "printer",numdocs,doc1,doc2,...,docn, layout,space)]

✓JR √Viewer

Prints multiple documents to the same page.

*printer* Printer description, exactly as it appears in the Control Panel Printers applet, e.g. "HP LaserJet 6P/6MP PostScript". If omitted, as in

FilePrintMultiple("",4,1,2,3,4,2,0.25,0.25)]
then the default printer will be used.

*numdocs* Number of documents to print on the same page.

*doc<x>* Document indices (1-32) for each of the *numdocs* documents to print. These indices are in the same order as the documents shown on the Window menu.

**layout** Arrangement of documents on the page. 0=vertical, 1=horizontal, n (n≥2)=n columns.

**space** Minimum space between graphs on the printed page, in inches.

The entire page minus margins is divided into equal areas with *space* inches between each area for each document (the area on the page devoted to each document will be equal, regardless of <u>Extents/</u><u>Intervals/Size</u> settings).

You can leave blank areas on the printed page by specifying a non-existant document index. For example:

FilePrintMultiple("",4,1,100,3,4,2,0.25,0.25)]

If there is no document with index=100 (and there never will be since the maximum number of open documents is 32) then this command will print document 1 in the upper left corner, 3 in the lower left corner, 4 in the lower right corner, and the upper right corner will be blank.

### 5.2.115 FileSaveAs command

Saves the document to *filename* as:

### Type Format

- 1 DPlot file
- 2 ASCII data, selected curve only
- 4 Comma-separated values
- 5 Unformatted data, 32-bit
- 14 Tab-separated values
- 15 Compressed DPlot file
- 16 Unformatted data, 64-bit
- 4096 Windows metafile

4097 Windows enhanced metafile

For more information on the file types see the description of the <u>Save As</u> menu command. If the " *Plugin Description*" form is used, the descriptive string must exactly match the description shown in the File Format dialog box, e.g. "BMP Picture".

### Metafile notes

DPlot will **not** save a metafile or an enhanced metafile of a 3D view or a 2D view of a surface plot with the "Type" set to "Shaded bands" (<u>ContourMethod(0)</u>) or "Both lines and shades" (<u>ContourMethod(6)</u>).

### Default Filename

If the *"filename"* argument is left blank or is a filename extension only, the plot title (w/o filename extension) will be used instead. For example, if this macro command:

FileSaveAs("Portable Network Graphics",".png")

is used and the plot title (the text appearing in the title bar) is "ex01.grf", then a PNG file named ex01. png will be saved.

### **Replaceable Parameters**

Starting with version 1.9.2.9, the *filename* may include replaceable parameters that are replaced at runtime by one of the title lines from the plot. |TITLE1| is replaced by the first title line, |TITLE2| by the 2nd, and |TITLE3| by the 3rd. These parameters must be in uppercase. For example, this sequence:

```
FileOpen("ex01.grf")
FileSaveAs(1,"c:\mydata\|TITLE1|.grf")
```

Saves the first example plot to the filename "Bicycling Caloric Expenditures.grf" to the c:\mydata folder.

Characters from the replacement string that would be illegal for usage in a filename ( $\langle :??"<> \rangle$ ) are replaced with an underscore character (_).

Within a macro only, if saving a file that was opened with a <u>FileOpen</u> command (as opposed to <u>ForFilesIn</u>), the filename may include replaceable parameters |PATH|, |FILENAME|, and |EXT|. These will be replaced at runtime by the corresponding elements from the last file opened via FileOpen. For example, this sequence:

```
FileOpen("sinewave.csv")
FFT()
Activate("FFT")
FileSaveAs(2,"|FILENAME|FFT|EXT|")
```

reads the file sinewave.csv, performs an <u>FFT</u> on the data and places the results in a new document, activates that new document, then saves the results to the file sinewaveFFT.csv.

The *filename* argument may include the shortcuts **|DPLOTDRIVE|**, **|DPLOTPATH|**, or **| DPLOTDOCS|** to set the destination drive or full path, respectively, to where DPlot is installed or, for **|DPLOTDOCS|**, to My Documents\DPlot. This is most often useful when DPlot is installed to a removable drive. All forms must be uppercase and surrounded by the pipe symbol (|) or will be ignored. Please note that the **|DPLOTPATH|** destination folder may not be acceptable to non-admin accounts on Windows XP and all accounts on Windows Vista and newer versions, depending on where DPlot is installed.

The trailing colon will be included with **|DPLOTDRIVE|**; similarly the trailing backslash will be included with **|DPLOTPATH|**. The drive letter and colon are included in **|DPLOTPATH|**; there is no need to use, for example, "**|DPLOTDRIVE||DPLOTPATH|***filename.grf*".

### Macros Only

**T**. ....

If placed in a ForFilesIn... NextFile loop, then only a filename extension, preceded by a dot, is used. **Exception:** if the file specification contains a | (pipe) symbol (see **Replaceable Parameters** above), then the entire filename is passed to DPlot as is and the filename extension (if any) should be included.

The filename used to save the document will be identical to the original filename but with the extension specified. If omitted, an extension dependent on the *type* code is automatically selected:

тур	EXTENSION	
е		
1	.GRF	
2	.ASC	
5	.BIN	
15	.GRFZ	
409	.WMF	
6		
409	.EMF	
7		

Extension

Filename extensions must be specified if the "Plugin Description" form is used.

### Example:

```
ForFilesIn("ex*.grf")
  FileSaveAs(4096,"")
  FileSaveAs(1,".new")
  FileSaveAs("BMP Picture",".bmp")
  FileClose()
NextFile
```

Saves all of the files matching the specification "ex*.grf" as Windows metafiles with the filename extension ".wmf", as DPlot files with the extension ".new", and (if the BMP File Export plugin is present) as RLE-encoded bitmaps with the extension ".bmp".

### Example:

```
ForFilesIn("ex*.grf")
   FileSaveAs("BMP Picture","c:\temp\|TITLE1|.bmp")
   FileClose()
NextFile
```

Saves all of the files matching the specification "ex*.grf" as RLE-encoded bitmaps in the c:\temp folder with filename equal to the plot title and a ".bmp" extension.

The dimensions and resolution of bitmap image files ("BMP Picture", "JPEG Picture", etc.) may be specified with the <u>SetPluginImageDims</u> command

### 5.2.116 FileType command

### [FileType(n)]

or

✓JR √Viewer

### [FileType("Plugin Description")

Sets the file type code prior to opening a document with the <u>FileOpen</u> command. *n* is one of:

- 1 DPlot file
- 2 ASCII file, one data set with arbitrarily spaced points.
- 3 ASCII file, one data set with evenly spaced points.
- 4 ASCII file, multiple columns.
- 5 Unformatted data, 32-bit floating point values.
- 6 Binary file produced by Pacific Data Model 9820 recorders, DNA/ Bendix format.
- 7 Binary file produced by Pacific Data recorders, **OLD** format.
- 8 Binary file produced by Pacific Data recorders, NEW format.
- 9 Nicolet Time Domain files (binary) created with the Nicolet System 400 Digital Oscilloscope.
- 10 Hardened Data Acquisition System (HDAS) files.
- 11 3 columns of randomly-spaced 3D points (X,Y,Z)
- 13 1D amplitudes (opened as a Box and Whisker plot by default)
- 14 ASCII text file, tab-separated columns
- 15 Compressed DPlot file
- 16 Unformatted data, 64-bit floating point values.
- 267 Identical to 11, but interpreted as a 3D scatter plot rather than a surface plot.

For more information on file types, see the description of the <u>Open menu command</u>. If the "*Plugin Description*" form is used, the descriptive string must exactly match the description shown in the Select File Type dialog box, e.g. "WAV Audio".

### 5.2.117 FillBetween command

|,"Legend"|,"LegendBelow"||]

Fills the area between Curve1 and Curve2 with a hatch pattern of Style and color R,G,B.

### Curve2

If *Curve2*=0 or is equal to *Curve1*, then the region formed by closing *Curve1* (connecting the first and last points) is filled.

If Curve2=-1, then the area between Curve1 and bottom edge of the graph is filled.

If *Curve2*=-3, then the area between *Curve1* and top edge of the graph is filled.

If *Curve2*=-4, then the area between *Curve1* and left edge of the graph is filled.

If *Curve2*=-5, then the area between *Curve1* and right edge of the graph is filled.

If *Curve2*=-6, then the area below *Curve1* will be filled to the bottom edge of the graph, but **only** for points in *Curve1* less than or equal to the *yref* value.

If *Curve2*=-2, then *yref* specifies the constant Y value for a reference line that serves as a boundary for the fill pattern. In this case *R2*, *G2*, *B2* specify the color components to use for areas below Y=*yref*.

Optional parameters "*Legend*" and "*LegendBelow*" are the legend entries for the fill areas. "*LegendBelow*" is only meaningful for *Curve2*=-2. You cannot include "*LegendBelow*" without "*Legend*", though you may specify a blank "*Legend*" with two successive quotation marks. Fill area legend entries will always follow the legend entries for curves, if present. Valid values for *Style* are:

- -1 Solid
- 0 Horizontal lines
- 1 Vertical lines
- 2 45 degree downward
- 3 45 degree upward
- 4 Horizontal-vertical crosshatch
- 5 45 degree crosshatch

The *mask* setting is optional and is equivalent to the Copy/Mask selections on the <u>*Fill Between*</u> <u>*Curves*</u> dialog. 0=Copy, 1=Mask. If *mask* is included, then *yref, R2, G2, B2* must also be included, though they may be blank entries (represented by adjacent commas.

Numeric curve indices may be replaced by substituting LegendHas

("substring1","substring2","substring3","). When this feature is used, DPlot will search for a curve whose legend entry contains substring1 and optionally substring2 and substring3. The index of the first curve whose legend matches that criteria will be substituted for LegendHas(...). If no match is found, no substitution is performed. Searches are case-specific.

For example, [SelectCurve(LegendHas("ABC", "GHI"))] will search for a curve whose legend contains both ABC and GHI.

"ABCDEFGHI" is a positive match; "ABCDEFGH" is not. Neither is "abcdefghi".

LegendHas(...) may be used in any command in place of a curve or data set index.

### See also

Fill Between Curves menu command

### 5.2.118 FillStyle command

### [FillStyle(Curve,Style)]

Sets the pattern used to fill the interior of bar charts in data set *Curve*. *Style* should be one of: 0 Hollow

- 1 Solid
- 2 Horizontal lines
- 3 Vertical lines
- 4 45-degree lines, sloping down from left-to-right
- 5 45-degree lines, sloping up from left-to-right
- 6 Horizontal-vertical crosshatch
- 7 45-degree crosshatch

### See also

BarChartStyle macro command

### 5.2.119 FindIntersections command

### [FindIntersections(*Curve1,Curve2,Options*)] XJR XViewer

Finds all intersections of *Curve1* and *Curve2*. *Options* is a combination of

- 1 Add data points at the intersections if necessary.
- 2 Add a label at each intersection. Ignored if the 1 bit is not also set.

Both *Curve1* and *Curve2* must have monotonically increasing values of X. If necessary (and appropriate), you can use the <u>EditSort</u> command prior to using **FindIntersections** to sort the data points of each curve in increasing X order.

✓JR XViewer

### 5.2.120 FontPoints command

### [FontPoints(index,points)]

Sets the size of the *indexed* font to *points*. Font indices used by DPlot are:

- 1 Numbers along the axes
- 2 First title line
- 3 Second and third title lines
- 4 X axis label
- 5 Y axis label(s)
- 6 Legend
- 7 Note(s)
- 8 Z axis label

For more control over the fonts, see the TextFont command

### 5.2.121 For AllDocuments command

### ForAllDocuments <u>Macros Only</u> XJR √Viewer

Allows you to loop through a set of commands for all open documents. "ForAllDocuments" must have a matching "NextDocument" statement, which terminates the loop and activates the next document in the sequence. "ForAllDocuments...NextDocument" loops may not be nested. The following example changes all fonts in all documents to red:

```
ForAllDocuments
   TextFont(1,,,,255,0,0)
   TextFont(2,,,,255,0,0)
```



```
TextFont(3,,,,255,0,0)
TextFont(4,,,255,0,0)
TextFont(5,,,255,0,0)
TextFont(6,,,255,0,0)
TextFont(7,,,255,0,0)
TextFont(8,,,255,0,0)
NextDocument
```

**Note:** A single SaveUndo command will only save an undo file for the currently active document. If you want the ability to undo changes made to each document by a ForAllDocuments loop, a SaveUndo command should be included at the start of the loop, as in:

```
ForAllDocuments
   SaveUndo()
   <other commands>
NextDocument
```

### 5.2.122 ForFilesIn command

ForFilesIn("*filespec"*) or

### ForFilesIn("filespec",A)

Allows you to loop through one or more operations on several files matching a given file specification. "ForFilesIn" commands must have a matching "NextFile" command, which terminates the loop. If the "A" parameter is included, then files are appended to the active document rather than opened in a new window (DPlot files may not be appended to each other; all other file formats may). "ForFilesIn...NextFIle" loops may not be nested. The following example opens each of the files in the current directory matching the file specification ex*.grf, prints the file, and closes it before operating on the next file matching that specification.

```
ForFilesIn("ex*.grf")
FilePrint()
FileClose()
NextFile
```

This sequence:

```
Directory("c:\mydata")
FileType(2)
ForFilesIn("*.dat",A)
NextFile
```

opens all files matching the specification "c:\mydata*.dat", assumes they are type 2 (ASCII, 1 curve per file) and appends them in the same document window.

Alternatively, the ForFilesIn command can read a list of filenames from a text file by placing an "@" symbol before the file specification. For example,

```
ForFilesIn("@file.lst")
   FilePrint()
   FileClose()
NextFile
```

reads a list of filenames from the file FILE.LST, prints and closes each file.

Macros Only

### 5.2.123 FunctionOfXY1Y2 command

### [FunctionOfXY1Y2("function",Xstart,Xstop,dX)] XJR Viewer

Generates X,Y pairs between X=*Xstart* and *Xstop* with spacing *dX* using a function of X and/or one or more Y values from existing curves. Y values from existing curves are included in the equation as "Yn", where n is the index of the curve. You are limited to the first nine existing curves (Y1, Y2, ... Y9). Curves used in the equation must have monotonically increasing X values. Note that this command is a superset of the simpler  $\underline{Y=f(X)}$  menu command; the function is not necessarily dependent on an existing curve.

The *Xstart*, *Xstop*, and *dX* arguments are optional. If *Xstart* and *Xstop* are not included and the function references one or more existing curves (Y1, Y2, etc.) then the overlapping extents of all referenced curves are used. If no existing curves are referenced but the plot **does** already have one or more curves, then the minimum and maximum X values for **all** curves are used for *Xstart* and *Xstop*. If the document is currently empty and *Xstart* and *Xstop* are not specified, then 0.0 and 1.0 are used. If *dX* is not specified and the document already contains one or more curves, then the minimum of the number of points for all existing curves, spaced evenly between *Xstart* and *Xstop*. If the document is empty, then the current maximum number of points (set with FileArrays) will be used.

For function syntax see the description of the  $\underline{Y=f(X)}$  menu command and the list of <u>available</u> functions.

See also

Y=f(X,Y1,Y2,...) menu command

### 5.2.124 FXY1Y2_XFrom command

### [FXY1Y2_XFrom(state,index)]

### XJR XViewer

This command controls how X values are generated using the <u>AddCurves</u>, <u>Compare</u>, <u>DivideCurves</u>, <u>FunctionOfXY1Y2</u>, <u>Lump</u>, <u>MultiplyCurves</u>, and <u>SubtractCurves</u> commands.

If **state** is non-zero, then X values for the output curve will be taken from the curve with the 1-based **index** (the i'th generated point will have an X value equal to the X value for the i'th point in curve **index**).

If **state**=0, then **index** is ignored. For FunctionOfXY1Y2 the specified start, stop, and interval values are used. For all other commands, the starting X value is set to the maximum starting X value for both curves referenced in the command; the ending X value is set to the minimum ending X value for those same curves. In all cases, the interval is set such that the number of points in the output curve will be the maximum of the number of points in all relevant curves, or the current maximum number of points if no curves are present in the case of FunctionOfXY1Y2.

See also

Y=f(X,Y1,Y2,...) menu command

### 5.2.125 GeneralOptions command

### [GeneralOptions(mask,flags)]

✓JR √Viewer

Sets the options found under <u>General Options</u>. The **mask** value controls which corresponding bits in the **flags** value are used. If a bit in the **mask** value is 0, the corresponding bit in **flags** is ignored. If a bit in the **mask** value is 1, the specified option is turned on if the corresponding bit in **flags** is 1, or off if that bit is 0. **Mask** and **flags** bits are:

0x0000001 Prompt to save changed plots 0x0000002 Allow undo 0x00000004 Always start maximized 0x00000008 Always auto-adjust scale factors on 3D plots 0x00000010 Always force symbols on/lines off for triangle plots 0x00000020 Antialias symbols and lines on display output 0x00000040 Initially turn symbols off/lines on 0x00000080 Recalculate extents of XY plots after hiding/unhiding curves 0x00000100 Adjust extents after applying transform to values 0x00000200 Show coordinates in popup window when mouse hovers over a data point 0x00000400 X values consisting of times are always sequential 0x00000800 Automatically read preference file based on filename 0x00001000 Assume input dates are of the form d/m/y 0x00002000 Disable zooming on 2D views with the mouse wheel. 0x00004000 Click and drag to zoom. For example, to turn off the "Always auto-adjust scale factors on 3D plots" setting, use [GeneralOptions(8,0)]. Mask and flags values may be expressed as hexidecimal or as integers. Multiple settings may be changed in one command; [GeneralOptions(0x00000030,0x00000020)] will turn on the "Antialias objects" setting and turn off the "Always force symbols on/lines off for triangle plots" setting.

### See also

General Options menu command

### 5.2.126 GenerateMesh command

### [GenerateMesh(*nx*,*ny*,*xll*,*yll*,*xur*,*yur*,*method*, XJR XViewer extrapolate,*np*)]

Generates a new document containing a rectangular grid of 3D points from the currently active 3D surface plot. For more information see the description of the <u>Generate Mesh</u> menu command.

nx, ny

Number of grid cells in the x and y directions.

xII, yll

Coordinates of the lower left corner of the generated grid.

xur, yur

Coordinates of the upper right corner of the generated grid.

method

=1 to use planar interpolation, 2 to use quadratic interpolation. 0, which in former versions used a (sometimes unstable) bivariate interpolation, will be replaced with 2 automatically.

extrapolate

If non-zero, will generate z values outside the existing triangular mesh (if necessary). This option has no effect if method=1.

np

Number of control points to use for quadratic interpolation. This value should be 3 or greater; 4 works well for generally smooth surfaces. Ignored if method=1.

### 5.2.127 GetPreferences command

### [GetPreferences("filename")]

Reads a preferences file (saved with the File>Save Preferences menu command or a <u>SavePreferences</u> macro command). If a path is not specified, the default preferences folder is assumed (%APPDATA%\DPlot\Preferences on Windows 2000 and all newer Windows versions). If a filename extension is not included, ".set" is appended.

✓JR √Viewer

### |DOCNAME|

If *filename* contains the text **[DOCNAME]** (including the enclosing pipe symbol |, and must be uppercase), **[DOCNAME]** will be replaced by the name appearing in the title bar of the document, minus any filename extension if present. Generally the document title is the name of the first file opened for the document or the name of a DPlot file if you save the document to that format, or **Plot1**, **Plot2**, etc. if a file has not been opened or saved. The document title may be changed with a <u>Caption</u> command.

For example if the document caption is [mydata.csv], then:

"|DOCNAME|" will be replaced with "%APPDATA%\DPlot\Preferences\mydata.set"

"c:\folder\|DOCNAME|" will be replaced with "c:\folder\mydata.set"

"c:\folder\|DOCNAME|.pref" will be replaced with "c:\folder\mydata.pref"

### 5.2.128 GrainSizeFlags command

### 

Controls the appearance of a grain size distribution plot. *flags* is a combination of:

- 0x00000001 Specify sieve size labels. If this bit is not set, bits 1 through 27 are ignored and all of the sieve sizes mentioned below will be drawn (if they will fit on the plot without overlapping and the plot extents include the given size)
- Bits 1-27 specify which sieve size labels to draw, dependent on bit 0x40000000 (British sizes, see below).

bit	U.S. sizes	British sizes
0x0000002	6 inch	125 mm
0x0000004	4 inch	75 mm
8000000x0	3 inch	63 mm
0x0000010	2 inch	50 mm
0x0000020	1-1/2 inch	37.5 mm
0x0000040	1 inch	28 mm
0x0000080	3/4 inch	20 mm
0x00000100	1/2 inch	14 mm
0x0000200	3/8 inch	10 mm
0x00000400	#3	8 mm
0x00000800	#4	6.3 mm
0x00001000	#6	5 mm
0x00002000	#8	3.35 mm
0x00004000	#10	2.36 mm
0x00008000	#12	2 mm
0x00010000	#16	1.7 mm
-------------	------	----------
0x00020000	#20	1.18 mm
0x00040000	#30	0.850 mm
0x0008000x0	#40	0.600 mm
0x00100000	#50	0.425 mm
0x00200000	#60	0.300 mm
0x00400000	#70	0.250 mm
0x00800x0	#100	0.212 mm
0x01000000	#140	0.150 mm
0x02000000	#200	0.125 mm
0x04000000	#270	0.075 mm
0x0800000	#400	0.063 mm

0x10000000 Draw bold vertical lines at material-type boundaries 0x20000000 Use <u>Udden-Wentworth</u> scale rather than <u>AASHTO</u> scale 0x40000000 Use British standard sieve sizes rather than U.S. sizes. 0x80000000 X axis increases from left-to-right.

If bit 0x40000000 of *flags* is set (use British sizes) **OR** bit 0x00000001 (specify sieve sizes) is **not** set, then the optional *other* parameter is ignored. Otherwise *other* specifies additional ASTM sieve sizes to draw that are common in the mining industry:

bit	U.S. sizes
0x0000001	#5
0x0000002	#7
0x0000004	#14
0x0000008	#18
0x0000010	#25
0x00000020	#35
0x00000040	#45
0x0000080	#80
0x00000100	#120
0x00000200	#170
0x00000400	#230
0x00000800	#325
0x00001000	#450
0x00002000	#500

#### Udden-Wentworth scale

A geometric scale of grain sizes which classifies particles of siliciclastic sediment from 4096 millimeters (boulders) in size down to 0.00006 millimeters (clay). This scale is almost universally accepted by modern sedimentologists (a type of geologist). It was first proposed in 1898 by Udden and then modified and extended in 1922 by Wentworth.

<u>Source</u>

# AASHTO

American Association of State Highway and Transportation Officials 444 North Capitol Street, N.W. Suite 249 Washington, D.C. 20001



# 5.2.129 GrainSizeFlagsEx command

#### [GrainSizeFlagsEx(flags)]

Specifies which user-defined sieve sizes are used. Those user-specified sizes might be pre-existing

✓JR XViewer

or be established with a GrainSizeOther command. This setting is ineffective unless bit 1 of the setting made with GrainSizeFlags is set (or the "Specify sieve size labels" box on the Grain Size Distribution Plot Options dialog is checked). flags is a combination of:

0x00000001 Use the first user-specified sieve size. 0x0000002 Use the second user-specified sieve size. 0x00000004 Use the third user-specified sieve size. 0x0000008 Use the fourth user-specified sieve size.

Combine *flags* options by adding them.

# 5.2.130 GrainSizeOther command

## 

#### "|, size3," text3"|, size4," text4"|||)]

Specifies from 1 to 4 custom sieve sizes to use on grain size distribution plots. size is the sieve size in millimeters, text is the label that will be used for this sieve. Note that these sieve sizes will not be drawn unless the corresponding bits in a GrainSizeFlagsEx command are set, or the corresponding checkboxes on the Grain Size Distribution Plot Options dialog are checked.

## 5.2.131 GridLines command

#### [GridLines(style)]

✓JR XViewer

Controls the appearance of the X and Y axes. Style=0 for axes only, =1 for a box around the plot with horizontal and vertical grid lines at each tick mark, =2 for a box around the plot with no grid lines. =3 for a box around the plot and vertical grid lines, horizontal tick marks, or =4 for a box around the plot with horizontal grid lines and vertical tick marks.

See also

Grid Lines or Box Options menu command

# 5.2.132 GridLinesOnTop command

# [GridLinesOnTop(option)]

For XY plots, draws grid lines below all other objects for option=0, above fill patterns for option=1, above fill patterns and curves for option=2, or above fill patterns, curves, and reference lines for option=3. Grid lines will never obscure an opaque legend, point label, or notation.

✓JR XViewer

See also

Symbol/Line Styles menu command

# 5.2.133 GridLineType command

#### [GridLineType(index,type)]

Sets the line style for the major (index=1) or minor (index=2) grid lines to type, where type may be 1 (solid lines) through 7.

types:

1									
2									-
3	• • • •		• • • •	• • • •	• • • •	• • • •	• • • •	••••	••
4									-
5	—	—	—	—	—	—	—	—	-
6	<u> </u>	<u> </u>			. <u> </u>	<u> </u>		<u> </u>	-
7									-

See also

Symbol/Line Styles menu command

# 5.2.134 HelpContents command

[HelpContents()]

Displays the Contents topic of this Help file.

# 5.2.135 HelpSearch command

#### [HelpSearch("string")] ✓JR √Viewer

Displays the Help topic found in the keyword list that matches the keyword string. If there is no match, displays the Search dialog box.

# 5.2.136 HideCurve command

# [HideCurve(index)]

Makes one or more curves invisible. This feature might be useful for temporarily hiding curves on a crowded plot. If index is greater than 0, then the index th curve is made invisible. If index is less than 0, then all curves except the - index th curve are made invisible. If index=0, all curves in the

✓JR √Viewer

✓JR XViewer

document are made invisible. Also see the ShowCurve command.

# 5.2.137 Histogram command

#### [Histogram(interval)] XJR XViewer

Plots the frequency distribution of the amplitudes of the curve selected with the SelectCurve command. The sampling interval is specified by *interval*. This command generates a new document. For more information see the Help topic on the Histogram command.

#### Macro Example

```
Histogram(100)
Activate("Histogram")
FillStyle(1,1)
```

See also

Histogram menu command

# 5.2.138 Independent Ticks command

```
[IndependentTicks(state)]
```

✓JR XViewer

For a non-zero state, with multiple dependent X or Y axes, if grid lines are used then tick marks on the second axis will be drawn at "nice" intervals, independent of the primary axis. If state is 0, then no additional tick marks will be drawn, and the labels on the second axis will be a function of the primary axis. This setting is applicable to linear and logarithmic scales.

Example:

IndependentTicks(0)

#### IndependentTicks(1)



# 5.2.139 InfoInterpolate command

# [InfoInterpolate()]

✓JR √Viewer

Equivalent to selecting the Interpolate menu command.

300

150

125

100

75

50

25

n

1000

900

Centimeters

270

# 5.2.140 InfoScan command

[InfoScan()]

✓JR √Viewer

Equivalent to selecting the Scan X, Y menu command.

# 5.2.141 InsertImage command

#### 

This command allows you to specify an image (BMP, EMF, GIF, JPG, PNG, or TIF) that will be drawn on the plot. Enhanced metafiles **must** have an extension of ".emf" to work properly with this command. Each plot may have up to 4 embedded images.

If the "*filename*" argument does not include a path, DPlot will look for it in My Documents\DPlot\Images. If the image does not reside in that folder then "*filename*" should include the full path to the file. All other parameters are optional.

flags0x0001 = X,Y in data space. If not set, X and Y specify the offset from the top left<br/>corner of the plot to the image anchor point, in inches.<br/>0x0002 = Horizontally centered on the anchor point<br/>0x0004 = Right-aligned on the anchor point<br/>0x0008 = Middle-aligned vertically on the anchor point<br/>0x0010 = Bottom-aligned on the anchor point<br/>0x0020 = Draw in background (before any other objects). If not set, the image is drawn<br/>after all other objects.

Combine *flags* options by adding them.

width, height Width and height of the image, in inches.

*transparency* Global transparency value applied to entire image. Valid range is 0 (invisible) to 255 (opaque). Default value = 255 (opaque). The transparency setting is ignored for enhanced metafiles.

Images are only applicable to 2D views of your data.

#### See also

Insert Image menu command DeleteImage macro command

# 5.2.142 InsertLink command

#### 

Inserts an active link from a DDE server into the current document. For more information see the <u>Insert Link</u> menu command.

# 5.2.143 InsertXY command

#### [InsertXY(*index,xnew,ynew*)]

Adds a data point to the curve selected with the <u>SelectCurve</u> command at the 1-based *index*'th position. If *index* is greater than the current number of points in this curve, the new point is simply appended to the end of the curve as with the <u>XY</u> command. Returns an error for *index* < 0. If this operation would cause the number of points in the curve to exceed the array sizes (set with

FileArrays), DPlot returns DDE FNOTPROCESSED to a DDE client, or causes an error in the DPlot macro facility.

## 5.2.144 Integrate command

#### [Integrate()]

# XJR XViewer

Integrates the curve or curves selected with the SelectCurve command. For more information see the Help topic on the Integrate menu command.

#### 5.2.145 InvertAxis command

#### [InvertAxis(axes)]

Inverts the scaling of XY plots on the X (axes=1) or Y (axes=2) axis, or both axes for axes=3. Normal scaling is restored for **axes**=0. This command performs the identical function as the *Invert X* and Invert Y checkboxes on the Extents/Intervals /Size dialog box. This command is ignored for polar charts, tripartite grids, grain size distribution plots, and triangle plots. NOTE: For 3D plots, this same functionality is accomplished by using negative scale factors with the ContourScales command.

# 5.2.146 LabelDirection command

#### ✓JR XViewer [LabelDirection(*dirx*, *diry*|, *dirz*|)]

Controls the orientation of the numbers along each axis. 0=horizontal, 1=45 degrees, 2=vertical. These settings are ignored for polar plots, triangle plots, and 3D views of 3D or 4D data.

Also see NumberFormat.

## 5.2.147 Legend command

#### [Legend(curve, "string")]

Sets the legend for curve number *curve* to the character string *string*. Use *curve*=0 to set the legend title.

You can build on the current legend or use the legend from a different curve with the \$LEGEND(n) placeholder, where n is the index of the curve whose legend entry you want to use. If (n) is omitted, the current legend for *curve* is used. For example:

#### [Legend(2,"\$LEGEND(1)<additional text>")]

will append <additional text> to the legend of curve 1 and use this as the legend for curve 2.

## 5.2.148 LegendParams command

#### [LegendParams(linelength,maxrows,frame, $\checkmark$ JR ×Viewer opaque,x,y,nSymbols,alignment,bkcolor)]

Sets various parameters for the legend. All parameters must be separated by commas. All parameters are optional; omitted parameters are denoted by two successive commas with no intervening spaces.

length of the line segment (in inches) appearing beside each legend string. linelength maximum number of legend strings to place in one column. maxrows *frame* = 1 to draw a rectangle around the legend.

# ✓JR XViewer

#### © 2014 HydeSoft Computing, LLC

- = 0 for no frame.
- opaque = 1 to hide all objects intersecting the legend.
  - = 0 for a transparent legend.
    - the location of the legend, expressed as a percentage of the plot window size (0.0 to 1.0).
- **nSymbols** number of symbols to use in each legend entry, when symbols are used. Limited to 0-3, inclusive.
- *alignment* defines how the legend is aligned relative to the specified location. Combination of horizontal and vertical alignment codes:

HorizontalVertical

- 0 Left 0 Top
- 1 Center 4 Middle
- 2 Right 8 Bottom
- *bkColor* sets the background color of the legend. If omitted or set to -1, the legend background will be the same color as the plot background (set with <u>BkColor</u>). This value is ignored if the legend is not opaque.

#### Examples:

X,V

#### [LegendParams(1.5)]

sets the length of the line segments used in the legend to 1.5 inches.

#### [LegendParams(,,1,,,,1)]

draws a frame around the legend and sets the number of symbols used to 1.

#### [LegendParams(,,,,,,10)]

sets the alignment point to the lower right corner of the legend.

#### [LegendParams(,,,1,,,,0x00FFFFC0)]

makes the legend background opaque, filled with a light cyan.

## 5.2.149 LegendShow command

Hides (*show*=0) or makes visible (*show*=1) the legend.

# 5.2.150 LineType command

#### [LineType(*curve,type*)]

**√**JR **×**Viewer

Sets the line style for curve number *curve* to *type*, where *type* may be 0 (no lines) through 7. If *curve* is set to -1, this command changes the line style used by **all** curves.

types:

1 2 3 4 5 5 6 7 See also Symbol/Line Styles menu command

# 5.2.151 LineWidth command

#### [LineWidth(*curve,width*)]

Sets the width of the line segments used to draw curve number *curve* to *width* 1000'ths of an inch. If *curve*=-2, *width* is used for all curves. If *curve*=-1, *width* specifies the line width used for the coordinate axes. If *curve*=0, *width* is the line width used for grid lines and tick marks; if=-3, *width* is the line width used for grid lines and tick marks; if=-3, *width* is the line width used for grid lines and tick marks; if=-3, *width* is the line width used for grid lines and tick marks; if=-3, *width* is the line width used for grid lines and tick marks; if=-3, *width* is the line width used for grid lines and tick marks; if=-3, *width* is the line width used for grid lines (those drawn as a result of <u>DenseGridLines</u>).

✓JR XViewer

For 3D data, *curve*=1 is used to set the line width for major contour lines, *curve*=2 is used for minor lines. The line width used for wireframe plots is set with *curve*=-4.

#### See also

Symbol/Line Styles menu command

## 5.2.152 ListPeaksOptions command

#### [ListPeaksOptions(options)]

Sets the options for the next time you use the <u>List Peaks</u> command on the Info menu or request peak values using <u>DPlot_Request</u> with the ListPeaks document topic. *options* is a combination of:

✓JR XViewer

options0x0001 = Minimum.<br/>0x0002 = Maximum.<br/>0x0004 = Maximum - Minimum.<br/>0x0008 = Mean.<br/>0x0010 = Standard deviation.<br/>0x0020 = Standard error.<br/>0x0040 = Median<br/>0x0080 = Skewness<br/>0x0100 = Kurtosis<br/>0x0200 = Number of points

See also

List Peaks menu command

## 5.2.153 LogTicks command

#### [LogTicks(option)]

Controls the frequency of tick mark labels on logarithmic scales. If **option=**0, labels are drawn only at powers of 10. If **option=**1, labels are drawn at every 5 times a power of 10. And if **option=**2, labels are drawn at all major tick marks. If **option=**3, labels are also drawn at minor tick marks (if minor tick marks are drawn).

### 5.2.154 Lump command

#### [Lump(Curve1,Curve2)]

Creates a new curve whose amplitudes are the sum of the amplitudes for *Curve1* and *Curve2*. This command has been kept for backward compatibility and is equivalent to the <u>AddCurves</u> command.

XJR XViewer

See also Y=f(X,Y1,Y2,...) menu command

# 5.2.155 ManualScale command

[ManualScale(xlo,ylo,xhi,yhi,zlo,zhi)] for 3D plots.

Forces the coordinates of the lower left and upper right corners of the plot. By default, DPlot automatically scales the plot. Recommended for real-time applications. The **ylo2**, **yhi2** parameters (XY plots) specify the extents of a second independent Y axis (established with the <u>Multiple Axes</u> menu command or with a <u>MultipleYAxes</u> command string. For 3D plots, **zlo** and **zhi** specify the Z axis extents.

All parameters are optional and **must** be separated by commas. Omitted parameters are represented by 2 successive commas with no intervening spaces. For example, to force the Y extents of an XY plot and use the existing X extents, use

#### [ManualScale(,ylo,,yhi)]

You may use equations in place of numeric entries for any argument. For example:

[ManualScale(=FLOOR(\$XMIN/100)*100,=FLOOR(\$YMIN/10)*10,=CEIL(\$XMAX/100) *100,=CEIL(\$YMAX/10)*10)]

sets the X extents to multiples of 100 that include the actual data extents, and the Y extents to multiples of 10.

As with the *Specify Extents* option of the <u>Extents/Intervals /Size</u> menu command, the actual extents used will be dependent on the tick mark interval on each axis. To ensure that the plot uses the specified extents, combine this command with a <u>TickInterval</u> command and set the extents to a multiple of the respective tick mark intervals.

#### See also

Extents/Intervals/Size menu command

# 5.2.156 Medians command

[Medians()]

#### XJR XViewer

Creates a new curve consisting of the median amplitude of all curves in a document

See also

Median of All Curves menu command

## 5.2.157 MenuHide command

[MenuHide()]

✓JR ✓Viewer

Hides the main application menu. If you use this command, you should also turn off the toolbar with the <u>ViewToolbar</u> command. Toolbar buttons and menu commands are closely linked and enabled/ disabled at the same times. If you do not hide the toolbar when hiding the menu, you'll most likely have disabled buttons that should be enabled, and vice versa.

# 5.2.158 MenuShow command

[MenuShow()]

Restores the main application menu after hiding it with the MenuHide command.

# 5.2.159 MenuUpdates command

#### 

If *state* is 0, DPlot's menu will not be updated/repainted. This command is primarily useful in realtime applications that send data to DPlot in short intervals using commands that normally would force the menu to be updated (like <u>SelectCurve</u>), sometimes resulting in annoying flashing. **Please note:** This state is persistent. If you turn off updates then you **must** eventually turn updates back on with [MenuUpdates(1)].

✓JR √Viewer

# 5.2.160 MoveCurves command

[MoveCurves("destination",index,move)] ×JR ×Viewer or

# [MoveCurves(DocNum,index,move)]

Moves (*move*=1) or copies (*move*=0) the *index'th* curve (or all curves for *index*=-1) from the reference document to the document with caption "*destination*" or, if the *DocNum* form is used, to the document with index *DocNum*. If *destination* is blank (or *DocNum*=0), the curve or curves are placed in a new document.

**NOTE:** For this command to work as expected with existing destination documents and the "*destination*" form, the destination document's caption must be known. DPlot will always generate a unique document caption, regardless of the use of the <u>Caption</u> command. For example, if your application uses the Caption command on two separate documents with the name set to "My New Plot" in both cases, DPlot will name the second document "My New Plot1". You can poll DPlot for a list of all document captions with this sequence of commands (Visual Basic):

Text1.LinkTopic = "DPlot|System"
Text1.LinkMode = 2
Text1.LinkItem = "Topics"
Text1.LinkRequest
Text1.LinkMode = 0

The Text1 text box will then be filled with a series of null-terminated strings. Two consecutive 0 bytes indicate the end of the string. The first and second strings will always be "XYPlot" and "System". The third and subsequent strings will be the captions of all open document windows. Given the list of all active documents, your application can then generate another caption for a new document that you know with certainty will be unique. You may then use the <u>FileNewEx</u> command to open a new document with that caption.

Alternatively, if the destination document was created via calls to <u>DPLOTLIB</u> functions, then the document index **DocNum** will have been returned by the function that created the document.

# 5.2.161 MultipleXAxes command

[MultipleXAxes(state,A,B)]

Controls the appearance of the X axis. For a non-zero **state**, a second X axis is drawn at the top of the plot with a scale determined by the relationship  $X2=A+B^*X1$ , where X1 refers to the primary X axis at the bottom of the plot and X2 to the second X axis. For **state**=0, no second X axis is drawn. The placement of the labels and/or tick marks on the second X axis is influenced by IndependentTicks.

Example:

MultipleXAxes(1,0,0.3048)



You may use equations in place of numeric entries for A & B. For example, if your input X consists of calendar dates and times with a constant interval of 30 minutes, then:

MultipleXAxes(1,=-\$XMIN*48,48)
NumberFormat(2,0)
IndependentTicks(1)

will produce a second X axis showing the 0-based index of the data point.

#### 5.2.162 MultipleYAxes command

[MultipleYAxes(type,A,B)]

✓JR XViewer

Controls the appearance of the Y axis.

<b>type</b>	Description
0	Single Y axis
1	Constants $A$ and $B$ are ignored. Y2=f(Y1)

2

A second Y axis is linked to the primary Y axis by the function Y2 =  $A + B^*$ Y1.

Independent Y2

A second Y axis is independent of the primary Y axis. Constant A = 1 to associate the selected curve with the left axis, = 2 to associate the selected curve with the right axis. Curve 1 **must** be associated with the left axis. Constant **B** is ignored.

For more information, see the Help topic on the Multiple Y Axes menu command.

**NOTE:** For *type*=2 and if used with <u>DPLOTLIB</u>'s DPlot_Command function, be sure to combine this command with a SelectCurve command, as in:

DPlot_Command("[SelectCurve(4)][MultipleYAxes(2,2,0)]")

This is necessary (as opposed to two separate calls to DPlot_Command) because each call to DPlot_Command causes several internal variables to be initialized. If these two commands are sent in different calls then the currently selected curve will be undefined.

For type=1, you may use equations for A & B. For example,

MultipleYAxes(1,=-\$YMIN/(\$YMAX-\$YMIN),=1.0/(\$YMAX-\$YMIN))
IndependentTicks(1)

will show a second Y axis with the values equal to 0 at the minimum Y value, 1 at the maximum Y value.

## 5.2.163 MultiplyCurves command

#### [MultiplyCurves(Curve1,Curve2)]

Generates a new curve with amplitudes equal to the amplitudes of *Curve1* times the amplitudes of *Curve2*.

XJR XViewer

X values for the new curve are taken from *Curve1*, subject to whether <u>FXY1Y2_XFrom</u> with a nonzero *state* value has been previously used. Linear interpolation is used to find the amplitude of *Curve2* at each X.

See also <u>Y=f(X,Y1,Y2,...)</u> menu command

# 5.2.164 NextDocument command

NextDocument Macros Only XJR √Viewer

Terminates a <u>ForAllDocuments</u>...NextDocument loop.

# 5.2.165 NextFile command

NextFile Macros Only XJR √Viewer

Terminates a ForFilesIn...NextFile loop.

# 5.2.166 NoteButtonText command

#### 

Specifies the text that is displayed on a button for button-activated text notations.

See also Add/Edit Note menu command

# 5.2.167 NoteOptions command

# [NoteOptions(flags)]

Changes the appearance of a notation. Flags is a combination of:

- 2 Draw a rectangular frame around the notation.
- 4 The notation is opaque; objects intersecting the notation will be hidden from view.
- 8 The notation is activated by a button.

# See also

Add/Edit Note menu command

# 5.2.168 NotePosition command

# [NotePosition(X, Y)]

Controls the position of the notation on the plot. X, Y are the coordinates of the upper left corner of the note, expressed as a percentage of the plot size (0.0 to 1.0).

#### See also

Add/Edit Note menu command

# 5.2.169 NoteTabstops command

[NoteTabstops(tab)]

Tab is the number of character cells used for tab positioning in a notation.

See also Add/Edit Note menu command

# 5.2.170 NoteText command

[NoteText("text")]

Specifies the text that is displayed in a notation.

See also Add/Edit Note menu command

# 5.2.171 NSEW command

## [NSEW(state)]

If state is non-zero, then values along the X or Y axis that use either Degrees, Minutes or Degrees, Minutes, Seconds number format will be displayed as absolute values with the appropriate suffix (N, S, E, or W). Positive values on the X axis will have the suffix E; negative values will have the suffix W. Positive values on the Y axis will have the suffix N, negative values will have the suffix S.

✓JR XViewer

✓JR XViewer

✓JR XViewer

**√JR X**Viewer

This setting is only used for the numbers along the X and Y coordinate axes and only for linear scales or Mercator projections.

See also

Number Formats menu command

### 5.2.172 NumberFormat command

#### 

Sets the style used to represent numbers along the X axis (**axis**=0), Y axis (**axis**=1), Z axis or secondary X axis on XY plots (**axis**=2), or secondary Y axis on XY plots (**axis**=3).

The optional argument *digits* specifies the number of digits to the right of the decimal place to display. If *digits* = -1, DPlot uses its default formatting: no trailing zeros, maximum of seven significant figures. This setting is ignored for amplitudes greater than or equal to 10,000,000 or less than 0.00001.

If the optional *thousands* argument is non-zero, DPlot will display the thousand's separator character before every 3rd digit in the whole number portion of a value. The separator character is set by Control Panel's Regional Settings applet.

The optional *direction* argument specifies the orientation of the numbers. 0=horizontal, 1=45 degrees, 2=vertical. *Direction* is ignored for secondary X or Y axes: DPlot will always use the setting for the primary X or Y axis, respectively. This setting is also ignored for polar plots, triangle plots, and 3D views of 3D or 4D data.

Valid values for style are:

- 0 default
- 1 Scientific notation
- 2 Fractions
- 3 Currency 1 (2 decimal places)
- 4 Currency 2 (whole numbers)
- 5 Time of day (hours)
- 6 Calendar date
- 7 Multiples of pi
- 8 Pi fractions
- 9 Month of the year
- 10 Day of the week
- 11 Powers of 10
- 12 None (no numbers are displayed)
- 13 Date and time
- 14 Hexadecimal
- 15 Percent
- 16 Degrees, Minutes
- 17 Degrees, Minutes, Seconds
- 18 Engineer
- 19 Elapsed time (HH:MM:SS)

20 Time of day (fractional days)

See also

Number Formats menu command

# 5.2.173 NumTicks command

#### [NumTicks(state,nx,ny,nz)]

✓JR XViewer

If *state* is non-zero, specifies the number of tick mark intervals that will be used on the X, Y, and Z (3D plots only) axes. If *state*=0, DPlot will automatically select the number of tick mark intervals.

All parameters must be separated by commas. *nx*, *ny*, and *nz* are optional. Omitted parameters are represented by 2 successive commas with no intervening spaces. If omitted then the current number of tick mark intervals for the respective axis will be maintained. For example, to set the number of tick mark intervals on the Y axis to 5 and leave the X (and Z for 3D plots) intervals alone, use

#### [NumTicks(1,,5)]

Alternatively, see the <u>TickInterval</u> command.

#### See also

Extents/Intervals/Size menu command

# 5.2.174 OHLC macro command

# [OHLC(open,high,low,close,options,color1, $\checkmark$ JR ×Viewer color2)]

Creates an open-high-low-close (OHLC) chart with the Open values taken from the 1-based **open** index, the High values from curve **high**, the Low values from curve **low**, and the Close values from curve **close**. The X values (normally dates) of the **open** curve will be used on the result, and the X values from the **high**, **low**, and **close** curves are ignored. That is, the n'th value from the **high** curve is used as the n'th High value for the Open curve, regardless of the actual **high** X values.

options is a combination of the following bits:

- 1 Use *color1* for *close* prices greater than or equal to *open* prices, or *color2* for *close* prices less than *open* prices. If not set, use the color associated with the *open* curve.
- 16 Candlestick chart.

None of **open**, **high**, **low**, or **close** indices may be the same. If any of the **open**, **high**, **low**, or **close** indices do not exist, then this command does nothing. If any of the **open**, **high**, **low**, or **close** curves are subsequently erased, then the OHLC settings for the **open** curve are erased.

Colors are in the form blue*65536+green*256+red, where blue, green, and red are between 0 and 255, inclusive. Colors may be specified in hexadecimal notation with a '0x' prefix. For example 0x00FFF00 is cyan (blue=255, green=255, red=0).

To turn off the OHLC setting for the *open* curve, use the OHLC command again with any of *high*, *low*, or *close* indices set to 0.

See also OHLC Chart menu command

# 5.2.175 OverlayDocument command

[OverlayDocument(docnum/,alpha/,

#### maskcolor//)] or [OverlayDocument("caption"|,alpha|, maskcolor//)]

Allows you to overlay the active document with the document specified by its index (*docnum* form) or its "*caption*". This is useful for overlaying contour plots with x,y data and bar charts with line charts, for example. The optional *alpha* parameter controls the transparency of the overlay. *alpha*=0 is completely transparent (invisible, in other words); *alpha*=255 is opaque (no transparency).

The *maskcolor* argument specifies a color (red+green*256+blue*65536, or hex 0x00bbggrr) in the overlay that will **not** be drawn if *alpha* is between 0 and 255. This is useful for making opaque portions of the overlay transparent: in particular an opaque legend box or contour line labels with an opaque background. This feature does not work particularly well with anti-aliased lines and/or smoothed screen fonts, since you can only specify a single color. Antialiased lines and fonts are generally less of a problem with printed plots.

The axis labels, numbers along each axis, and the axes themselves in the overlay plot will not be drawn. Extents are forced to match those of the overlaid plot. The scale type of the overlay is switched to that of the overlaid plot, with the exception that you may overlay a bar chart with a standard linear X, linear Y plot and vice versa.

This feature is **not** intended for use with 3D views of 3D/4D data at this time, though DPlot will not prevent you from using it on those plot types.

NOTE: This is a new beta feature (and so has no equivalent menu command in this version). We strongly suggest that you save your work before using this command, as some combinations of plot types and/or scales may be unstable. Transparency may or may not work on printed plots, depending on the printer - if you find that to be the case then you can of course save the plot to a bitmap image, and print that image from your favorite image-editing program.

# 5.2.176 Paste command

#### [Paste()]

# Pastes the contents of the Clipboard to the active document. The format of the Clipboard contents

must match that described under the <u>Paste command (Edit menu)</u> topic.

✓JR XViewer

✓JR XViewer

XJR √Viewer

# 5.2.177 PasteLink command

#### [PasteLink()]

Pastes the contents of the Clipboard to the active document and links the source data. Changes to the data in the source application will (assuming the source application can act as a DDE server) automatically be reflected in the destination plot. The format of the Clipboard contents must match that described under the <u>Paste command (Edit menu)</u> topic.

# 5.2.178 Pause command

Pause

e Macros Only

Pauses the macro and displays a MessageBox allowing you to continue or abort the macro. This macro command is useful for testing purposes.

# 5.2.179 PolarPlotStyle command

[PolarPlotStyle(flags)]

Changes the appearance of polar plots. Flags is a combination of:

- 0x0001 0 degrees is at the top of the plot, rather than the default right.
- 0x0002 Angles are measured clockwise rather than the default counterclockwise.
- 0x0004 Forces a circular plot.
- 0x0008 Radial axis labels are vertical.
- 0x0010 Radial labels are centered on the radius.
- 0x0020 0 degrees is labelled.
- 0x0040 Angles are displayed as 0 to  $\pm$ 180 degrees rather than the default 0 to 360.
- 0x0080 Transform points with negative radii to angle+180 degrees, radius=abs(radius). Note that unlike other options this operates on the actual data values and is therefore not reversible. The value added to the angle (X) coordinates is either 180 for angles in degrees or PI for angles in radians.
- 0x0400 Angles are in degrees rather than the default radians.
- 0x7000 Bits determining at which radial line the axial grid lines start.
  - 0x0000 Center of plot
    - 0x1000 1st radial line
    - 0x2000 2nd radial line
    - 0x3000 3rd radial line
    - 0x4000 4th radial line
    - 0x5000 5th radial line
    - 0x6000 6th radial line
    - 0x7000 7th radial line

Combine *flags* options by adding them.

See also

Polar Options menu command

# 5.2.180 PostponeRedraw command

## [PostponeRedraw()]

Delays redrawing any changes made to the document. This command may be useful when making many minor changes to a plot that would ordinarily cause the plot to be redrawn. The document will be redrawn after issuing a <u>ViewRedraw</u> command, or after issuing any command that causes the conversation to reference a different document (<u>FileOpen</u>, for example).

**NOTE:** The setting for PostponeRedraw will remain in effect only for the current DDE conversation. For macros, the entire macro is handled by a single conversation. However, if you are using <u>DPLOTLIB</u> function calls to communicate with DPlot, and specifically calling DPlot_Command, the DDE conversation is established and terminated for each call. If you have several commands for which you are calling DPlot_Command and want to postpone redrawing, group those commands together in one call.

# 5.2.181 Preserve3DScaleFactor command

#### [Preserve3DScaleFactor(state)]

For a non-zero or omitted *state*, this command forces DPlot to re-use the current scale factor and extents for 3D scatter plots and 3D contour plots when the plot is redrawn, regardless of any changes in the data extents or the viewing angles. The same scale will be used until a) the plot is closed, b) the user opens either the Contour Options or Scatter Plot Options dialogs, or c) this command is used with *state*=0. This command is useful for maintaining the scale when producing a series of frames that will subsequently be used to generate an animation. Note that the use of this

# ✓JR √Viewer

command may cause a plot to be clipped. To ensure that a plot in a series of viewing angle changes is **not** clipped, you can first walk through all frames that will be generated and find the frame that results in the smallest scale, then display the plot at those viewing angles before the use of Preserve3DScaleFactor.

#### Example:

```
char szCmd[128];
DPlot_Command(1,"[SetClientRect(1,100,100,739,579)][SetPluginImageDims
(640,480)]"
                "[SetImageCrop(0)][Preserve3DScaleFactor()]");
for(i=0; i<360; i+=2)
{
    sprintf(szCmd,"[ContourView(%d)]",i);
    DPlot_Command(1,szCmd);
    sprintf(szCmd,"[FileSaveAs(\"BMP Picture\",\"c:\\temp\\%03d.bmp\")]",
    i);
    DPlot_Command(1,szCmd);
}
DPlot_Command(1,szCmd);
}
```

This sequence sets the document window to 640x480 and produces bitmap images with the view azimuth varying from 0 to 358 degrees in steps of 2 degrees, saving those bitmaps to c:\temp. You can then use Windows Movie Maker to import those bitmaps and create an animation. This example (6Mb) on the DPlot web site was generated using the code above.

#### 5.2.182 ProbabilityFlags command

#### [ProbabilityFlags(flags)]

Changes the appearance of probability plots. *Flags* is a combination of:

- 0x0001 For probability scales on the X axis, do **not** draw an inverted scale at the top of the plot. The default is to draw that scale.
- 0x0002 For probability scales on the Y axis, do **not** draw an inverted scale on the right side of the plot. The default is to draw that scale.
- 0x0004 Input values range from > 0 (no chance) to < 1 (certainty). If not set, input values range from > 0 to < 100.

Combine *flags* options by adding them.

# 5.2.183 RefLine command

#### [RefLine(options,style,color,width,value/,"label"/)]

Creates a reference line with *value* = some constant X (XY plots only) or constant Y (XY or 1D plot) with line style *style* (0-7), *color* (red+green*256+blue*65536) and line *width* in 1000'ths of inches. The optional *label* will be added to the plot legend if used, or be drawn centered on the reference line if the 0x0004 bit of *Options* is set. *Options* is a combination of:

- 0x0001 Reference value is a constant Y, line will be horizontal. If not set, reference value is a constant X and the line will be vertical. For 1D plots this flag is implied. For polar plots if this flag is set, *value* specifies the radius of a circle. If not set, *value* is the angular value in degrees.
- 0x0002 internal use only
- 0x0004 Label (if included) will be drawn centered on the line rather than placed in the legend. This

# ✓JR ×Viewer

option is ignored for 3D data: the label will always be drawn centered on the line.

- 0x0008 Reference value is associated with the secondary X axis.
- 0x0010 Reference value is associated with the secondary Y axis.
- 0x0020 A black frame will be drawn around the label. This setting is ignored if the 0x0004 bit is not set.
- 0x0040 The label will be opaque. This setting is ignored if the 0x0004 bit is not set.
- 0x0080 The reference line and any associated legend entries will be hidden (not drawn).

Combine *options* by adding them.

# See also

Reference Lines menu command

# 5.2.184 RemoveLink command

# [RemoveLink("Application","Topic","Item")]

Unlinks the corresponding link(s) that were previously established by the <u>Insert Link</u> menu command or the <u>InsertLink</u> macro command. You can use the wildcard character "*" for either the "*Topic*" or "*Item*", but not the "*Application*". This command does **not** erase data from DPlot; it simply removes the association with the server application so that changes to the data in the server are not reflected in DPlot.

# 5.2.185 ReorderCurves command

# ReorderCurves(curve1,curve2,curve3,...)

Reorders the curves on an XY plot. This command is roughly equivalent to the **Reorder Curves** command on the **View** menu. The list of curves (**curve1,curve2,...**) is a list of integer indices of the current curve ordering. For example, **ReorderCurves(2,4,3,1)** will reorder the curves such that the curve that is currently second will be the first curve, the curve that is currently fourth will be second, etc.

# 5.2.186 Repaint command

# Repaint()

Similar to <u>ViewRedraw</u>, but simply redraws a saved bitmap of the plot if one is available. If a bitmap is **not** available for this document, this command is identical to ViewRedraw. This command is primarily useful in applications that that have used <u>SetOutputWindow</u> to set the output destination to some area within the calling application's window rather than the default DPlot document window. For example, an application might send a Repaint command to DPlot when it receives a WM_PAINT message due to the application window being covered/uncovered.

Please note that Repaint, unlike ViewRedraw, does **not** redraw a plot from scratch if a bitmap of the plot is available. Rather it simply does a BitBlt with the existing bitmap. So if your application uses <u>PostponeRedraw</u> followed by several commands that change the data or the plot in some way, then uses Repaint, you will almost certainly not get the result you expect; changes following PostponeRedraw will not be reflected in the new plot until a ViewRedraw command is issued.

# ✓JR ×Viewer

✓JR XViewer

✓JR XViewer

# the or

# 5.2.187 Request function

#### Request macro Macros Only XJR √Viewer

This function may be used to retrieve selected information from the currently active DPlot document, for substitution in other macro commands. Syntax:

#### VAR_x=Request("item")

where **x** is any of the uppercase letters from A to Z. VAR_x is case-specific: VAR_A will be interpreted as expected; var_a will not.

Valid *items* are shown below **Document Topics** in the <u>Transferring DPlot data to other applications</u> topic.

Variables will be replaced by the corresponding result in any subsequent macro command.

#### Example:

```
VAR_A=Request("ListPeaks")
TextNoteEx("VAR_A",0.5,0.5,6,5,8,0,0,-1)
```

will create a text notation centered horizontally and vertically at the center of the plot, containing the output of the List Peaks command.

#### Example:

```
VAR_A=Request("path")
VAR_B=Request("fname")
FileSaveAs("Portable Network Graphics","VAR_AVAR_B001.png")
```

If you open the DPlot file c:\my data\my dplot file.grf and run this macro, the macro will save a PNG file c:\my data\my dplot file001.png

#### 5.2.188 RunPlugin command

RunPlugin("Plugin Description")

RunPlugin("Plugin Description","Options")

Runs the plugin module on the Generate menu with description "*Plugin Description*". The descriptive string must exactly match the description shown on the Generate menu, minus any trailing ellipsis. For plugins that support it (*Moving Average*, for example) and DPlot version 2.3.2.1 or later, "*Options*" are a plugin-specific character string that will bypass the input dialog (if one exists) and use those options.

XJR XViewer

For more information on Plugins see the <u>Plugins</u> topic.

# 5.2.189 SavePreferences command

#### 

Reads a preferences file which may subsequently be read by the File>Get Preferences menu command or a <u>GetPreferences</u> macro command. If a path is not specified, the default preferences folder is assumed (%APPDATA%\DPlot\Preferences on all versions of Windows, Windows 2000 or newer). If a filename extension is not included, ".set" is appended.

#### |DOCNAME|

If *filename* contains the text **|DOCNAME|** (including the enclosing pipe symbol |, and must be

uppercase), [DOCNAME] will be replaced by the name appearing in the title bar of the document, minus any filename extension if present. Generally the document title is the name of the first file opened for the document or the name of a DPlot file if you save the document to that format, or Plot1, Plot2, etc. if a file has not been opened or saved. The document title may be changed with a Caption command.

For example if the document caption is [mydata.csv], then:

"|DOCNAME|" will be replaced with "%APPDATA%\DPlot\Preferences\mydata.set"

"c:\folder\|DOCNAME|" will be replaced with "c:\folder\mydata.set"

"c:\folder\|DOCNAME|.pref" will be replaced with "c:\folder\mydata.pref"

NOTE: If the Automatically read preference file based on filename option under the General command on the Options menu is checked, or if that flag is set with, for example, [GeneralOptions (0x00000800,0x00000800)], then when opening a data file for a new document DPlot will check for the existence of a preferences file (previously saved with this command) with the same filename and extension of ".set", first in the folder where the data file resides, then in the default preferences folder (%APPDATA%\DPlot\Preferences). If found, that preferences file will be processed. This might be a handy option if you have data files which frequently change, but you don't want to go through the same formatting steps each time you open that file. This option is not applicable to DPlot files, which by definition contain all of the formatting options that would normally be saved to a preferences file. When opening multiple files, this option is only applicable to the first file selected.

# 5.2.190 SaveUndo command

#### [SaveUndo()] Macros Only

If used, this command should **only** be inserted at the start of a macro, before any other commands. This command forces DPlot to save the current state of a plot so that the actions of the macro can later be undone with the Undo command on the Edit menu. This command has no effect if undo is disabled

# 5.2.191 Scale command

#### [Scale(code)]

Sets the scaling code for the plot (linear, logarithmic, etc.). The low byte of *code* should be one of the followina:

- Linear X, Linear Y 1
- 2 Linear X. Logarithmic Y
- 3 Logarithmic X, Linear Y
- 4 Logarithmic X, Logarithmic Y
- 5 Tripartite (log-log) scale
- 6 Linear X - Probability Y
- 7 Grain Size Distribution
- 8 Polar coordinates
- 9 Bar chart
- 10 Logarithmic X, Probability Y
- 11
- Probability X, Linear Y Probability X, Logarithmic Y 12
- Probability X, Probability Y 13
- 14 Triangle Plot
- 15 N1.85 Hydraulic scale
- 16 Mercator projection

#### ✓JR XViewer

The high byte of the scaling code contains scaling-specific information. Currently, this value is used only for tripartite grids, grain size distributions, and polar coordinates. For all other scale types, this value is reserved for future use and must be 0.

#### Tripartite grids

The high byte indicates what units to use for the grid. Supported values are:

Code

5	Hz, inches/sec, inches, and g's
261	Hz, feet/sec, feet, and g's
517	Hz, mm/sec, millimeters, and g's
773	Hz, cm/sec, centimeters, and g's
1029	Hz, meters/sec, meters, and g's
1285	Period (seconds), inches/sec, inches, and g's
1541	Period (seconds), feet/sec, feet, and g's
1797	Period (seconds), millimeters/sec, millimeters, and g's
2053	Period (seconds), centimeters/sec, centimeters, and g's
2309	Period (seconds), meters/sec, meters, and g's

#### Grain size distributions

The high byte indicates what units the grain sizes are in. Supported values are:

Cod

е	
7	Millimeters

263 Inches

#### **Polar coordinates**

The high byte indicates what units the angular dimensions are in. Supported values are:

Cod

- e 8 Ra
- 8 Radians 264 Degrees

# 5.2.192 ScatterPlotLines command

#### [ScatterPlotLines(state,width1,width2)]

#### ✓JR ×Viewer

For 3D scatter plots, state may have the following bits set:

- 1: Draws lines between data points.
- 2: Draws droplines from each symbol to the base XY plane.
- 4: Droplines are drawn in the symbol color. If not set, droplines are drawn in black. Ignored if (state & 2)=0.
- Lines between data points will be in the color of the first endpoint if the <u>ColorBasis</u> is set to 0 (colors based on Z value). If not set, these lines will be drawn in black. Ignored if (state & 1)=0.

Line width for line segments between points in thousandths of inches is specified by *width1*. Line width for droplines in thousandths of inches is specified by *width2*. All parameters are optional and must be separated by commas. Omitted values retain their previous setting.

#### Example:

[ContourView(30,20,2)]
[ScatterPlotLines(6,,30)]



See also <u>ColorBasis</u> macro command Symbol Styles menu command

# 5.2.193 SelectCurve command

#### [SelectCurve(n)]

#### ✓JR √Viewer

Selects the curve or curves to perform subsequent operations on. Default = 1 (first curve). If curve n does not currently exist, DPlot creates a new curve. If n exceeds the maximum number of curves (set with <u>FileArrays</u>) DPlot attempts to resize the arrays to accommodate a new curve. If n is set to -1, then any subsequent command that operates on one or more curves will operate on **all** curves. For multiple selections (but fewer than exist in the entire plot), use multiple curve indices separated by commas. For example,

#### [SelectCurve(1,3,5)]

will select curves 1, 3, and 5 for subsequent operations for commands that allow multiple curves to be operated on.

#### Differences between macros and calling DPlot_Command from another program

In the context of a macro the SelectCurve command may be a separate command from all subsequent commands (and indeed, must be). For example:

```
SelectCurve(3)
EditOperateY("Y*1000")
```

will select curve 3 and multiply the amplitudes of curve 3 by 1000.

But in the context of calling DPlot_Command from another program, this works differently. In C:

```
DPlot_Command(docnum,"[SelectCurve(3)]");
DPlot_Command(docnum,"[EditOperateY("Y*1000")]);
```

At first glance this would appear to be equivalent to the macro. But every call to DPlot_Command establishes a new conversation, sends the command, then terminates the conversation. Upon every new conversation DPlot resets several variables, including the selected curve number (to 1). This is done primarily to prevent multiple programs or the user from running over each other with unintended selections/operations.

In the case above, the EditOperateY command will operate on curve 1. In general if there is an association between one command and another, those commands should be grouped into one call, as in:

DPlot_Command(docnum,"[SelectCurve(3)][EditOperateY("Y*1000")]);

The command string for DPlot_Command is limited to 32K characters, so in almost all cases you can group as many commands into one call as you need.

Numeric curve indices may be replaced by substituting LegendHas

("substring1"],"substring2"],"substring3"]]). When this feature is used, DPlot will search for a curve whose legend entry contains substring1 and optionally substring2 and substring3. The index of the first curve whose legend matches that criteria will be substituted for LegendHas(...). If no match is found, no substitution is performed. Searches are case-specific.

For example, [SelectCurve(LegendHas("ABC", "GHI"))] will search for a curve whose legend contains both ABC and GHI.

"ABCDEFGHI" is a positive match; "ABCDEFGH" is not. Neither is "abcdefghi".

LegendHas(...) may be used in any command in place of a curve or data set index.

#### 5.2.194 SetBackgroundImage command

#### [SetBackgroundImage("filename",full_image, force_extents,left,top,right,bottom,foreground, transparency,bkColor)]

✓JR XViewer

This command allows you to specify an image (BMP, GIF, JPG, PNG, or TIF) that will be drawn within the plot extents. The background image will be stretched to fill the plot exents, regardless of its width/height relative to the plot width/height.

If the "*filename*" argument does not include a path, DPlot will look for it in My Documents\DPlot\Images. If the image does not reside in that folder then "*filename*" should include the full path to the file. All other parameters are optional.

If you want to **remove** an existing background image, supply a blank "*filename*", e.g. [SetBackgroundImage("")]. All other arguments are optional.

- *full_image* Default value is 1 (entire image will be used). If set to 0 then the *left, top, right*, and *bottom* values specify the top-down extents within the image that you want to use as a background. *left, top, right*, and *bottom* are ignored for a non-zero *full_image* value.
- force_extents Default value is 1. If non-zero then the plot extents will be adjusted such that the background has a 1:1 pixel ratio. For example if the background image is 800 x 600 pixels and this box is checked, then the width and height of the plot will be set to a 4:3 ratio. If <u>"Specify plot size"</u> was previously selected then the maximum dimension will be used for the larger of the width/height. If "Specify plot size" has not been previously selected then the maximum dimension will be set to 8 inches. This adjustment takes place at the time this command is issued and is not persistent: Subsequently changing the plot size with a <u>Size</u> command or with the <u>Extents/Intervals/Size</u> menu command may result in a distorted background image.

foreground If non-zero, the image will be drawn after all other objects. Default=0 (image is drawn

before all objects).

- *transparency* Global transparency value applied to entire image. Valid range is 0 (invisible) to 255 (opaque). Default value = 255 (opaque).
- **bkColor** Color that will **not** be drawn, in the form (*blue**65536)+(*green**256)+*red*, where *blue*, *green*, and *red* range from 0 to 255. The **color** value may be expressed as a decimal or hexadecimal value; hexadecimal notation should include a "0x" prefix. For example 0x00FF00FF and 16711935 both represent magenta (*blue=red=255*, *green=0*). This value will be ignored for 32-bit images with an alpha channel. If **bkColor** is set to a negative value, no color will be removed from the image.

Background images are only applicable to 2D views of your data, and are not used for <u>polar plots</u> or <u>triangle plots</u>.

#### See also

Background Image menu command

### 5.2.195 SetBitmapDims command

#### [SetBitmapDims(wPixels,hPixels)]

#### ✓JR √Viewer

Sets the size of bitmaps copied to the Clipboard by the Copy Picture command.

# 5.2.196 SetClientRect command

#### 

Resizes a document window or the main application window such that the client area (the work area of the window, not including borders, menu, caption, etc.) has the specified dimensions.

#### index

- -1 = document associated with the current DDE conversation.
- 0 = main application window.

1-32 = index of a child document window.

To set the client area to a specific width and height, use

[SetClientRect(index,left,top,left+width-1,top+height-1)]

#### 5.2.197 SetDecimalSymbol command

#### 

Sets the decimal symbol for numbers. If a comma is used, then multiple arguments to functions in  $\underline{Y=f(X)}$  and similar commands **must** be separated by semicolons. This setting has no effect on macros or on calls to <u>DPLOTLIB</u> functions from another program: The decimal symbol in that context must be a period '.', and arguments must be separated by commas. Text file output and saving data to the Clipboard is also not affected by this setting: DPlot will use a '.' for the decimal symbol and CSV files will use a comma as the column delimiter.

If **symbol** is omitted, DPlot will use whatever symbol is currently set with Control Panel. (This tie to Control Panel is **not** persistent; if the user later changes the Control Panel setting, the decimal symbol assumed by DPlot does not automatically change.)

**Please note:** This setting is global within DPlot and persists after your program is finished interacting with DPlot. If you need to use this command because, for example, your program instructs DPlot to read CSV files and your end users typically use a comma for a decimal symbol, then you should check this setting before forcing the decimal symbol, and restore the user's setting when done:

```
char sDecimal[2];
int dwSize=sizeof(sDecimal);
DPlot_Request(0, "DecimalSymbol", sDecimal, dwSize);
DPlot Command(0,"[SetDecimalSymbol(\".\")]");
sprintf(cmd,"[SetDecimalSymbol(\"%s\")]",sDecimal);
DPlot Command(0,cmd);
```

# 5.2.198 SetFilename macro command

#### ✓JR XViewer [SetFilename("filename")]

Allows you to specify a default filename for a document. If the user saves the file, this default filename will be presented in the File name text box on the Save As dialog. (Normally this text box is blank for documents that have never been saved.) The user may override the default filename, as with any other document. If *filename* includes a drive or path, that drive/path will be the default folder.

NOTE: SetFilename does not save a document. To save a file programmatically use the FileSaveAs command.

# 5.2.199 SetImageCrop command

# [SetImageCrop(crop)]

Specifies whether bitmap images (BMP, GIF, JPG, PNG, and/or TIF files) subsequently saved with a FileSaveAs command or retrieved with a call to the DPLOTLIB function DPlot_GetBitmap, et. al. should have white space cropped from the image. If crop is non-zero, all but a 10-pixel white space border will be cropped from the image. If crop is 0, the specified dimensions (set with SetPluginImageDims) will be used and no cropping is performed. The crop parameter for image files is saved and restored when DPlot is shut down/opened. For images generated by DPlot GetBitmap, the crop parameter is always set to 0 when DPlot is started.

# 5.2.200 SetMetafileDims command

# [SetMetafileDims(wInches,hInches)]

Sets the size of metafile pictures copied to the Clipboard by the Copy Picture command.

# 5.2.201 SetOutputWindow command

# [SetOutputWindow(*hwnd*,*left*,*top*,*right*,*bottom*)] $\checkmark$ JR ×Viewer

Instructs DPlot to render all subsequent graphics to the window with HWND handle hwnd as opposed to the normal document window. *left. top. right. bottom* specify the output rectangle in pixels, relative to the client area of the window. If *hwnd* is set to 0, normal rendering to the DPlot document window is restored.

# ✓JR XViewer

This command is not useful in macros, since there is no macro facility for retrieving window handles. It **can** be used, though, to send DPlot output to a window within another application that is controlling DPlot via <u>DPLOTLIB</u>. This command is primarily useful for real time applications that are updating a graph at a fast rate. If you want to create a plot from your application that will not be updated frequently, <u>DPlot_GetBitmap</u> or <u>DPlot_GetBitmapEx</u> are preferable to this method.

For best (fastest) results with real-time graphing you should use the <u>ManualScale</u> command with XY plots. If ManualScale is used, then whether data is added to the graph using an <u>XY</u> or <u>XYXY</u> command or by calling the <u>DPlot AddData</u> function in DPLOTLIB.DLL, only new line segments will be drawn. If ManualScale is **not** used (that is, you allow DPlot to determine good extents for the graph), then the entire graph will be redrawn each time it is updated. Another feature of ManualScale with real-time graphing is that if the X value is greater than or equal to the right extent of the graph, DPlot will shift the graph left by some fraction of the X extents. The default shift is 50% of the extents and may be controlled with <u>SetPanFraction</u>.

There are at least two consequences to using ManualScale:

- Since no bitmap is saved and the drawing is done directly to another application's window, if that window is covered/uncovered then the image will not be automatically updated as it would with a bitmap assigned to the properties of a picture box. You can handle this by forcing DPlot to redraw the entire graph with a <u>ViewRedraw</u> command whenever a paint message is received.
- 2) For performance reasons, new line segments will always be drawn using solid lines regardless of any <u>LineType</u> settings. This might be a bit disconcerting if a window is covered/uncovered, since the repainted portion of a graph will be drawn using the specified line styles, but new line segments will be drawn using solid lines. For this reason it is recommended to always use solid lines with real-time applications.

#### **Examples:**

In the following examples the calling application wants output drawn to a control owned by that application.

```
Visual Basic:

Type RECT

Left As Long

Top As Long

Right As Long

Bottom As Long

End Type

Declare Function GetClientRect Lib "user32" (ByVal hwnd As Long, lpRect As RECT) As

Long

.
```

Dim rcPic As RECT Dim sExec As String

```
ret = GetClientRect(Picture1.hwnd, rcPic)
sExec = "[SetOutputWindow(" + Str$(Picture1.hwnd) + "," + _
Str$(rcPic.Left) + "," + _
Str$(rcPic.Top) + "," + _
Str$(rcPic.Right) + "," + _
Str$(rcPic.Bottom) + ")]"
ret = DPlot_Command(DocNum, sExec)
```

#### C:

RECT rcFrame; char szExec[256];

GetWindowRect(GetDlgItem(hwnd,DLG_PICTURE),&rcFrame); MapWindowPoints(NULL,hwnd,(LPPOINT)&rcFrame,2);

sprintf(szExec,"[SetOutputWindow(%d,%d,%d,%d,%d)]", hwnd,rcFrame.left,rcFrame.top,rcFrame.right,rcFrame.bottom); DPlot Command(DocNum,szExec); C#: Rectangle rcPic; rcPic = pictureBox1.ClientRectangle; string cmds = String.Format("[SetOutputWindow({0},{1},{2},{3},{4})]", pictureBox1.Handle, rcPic.Left. rcPic.Top, rcPic.Right, rcPic.Bottom); dplot.DPlot Command(DocNum, cmds);

(where the dplot class is defined in dplot.cs in the various C# example folders)

# 5.2.202 SetPanFraction command

#### [SetPanFraction(fraction)]

Sets the pan distance for XY plots and 2D views of surface plots when 1) using the arrow keys to pan, 2) sending data to DPlot via the XY or XYXY commands, or 3) sending data to an existing plot with the DPlot AddData function in DPLOTLIB.DLL. Default=0.5 (half the plot width or height). Limits are 0 to 1.0. If set to 0, then the plot will not be panned by a call to DPlot AddData or with the various XY commands.

#### 5.2.203 SetPluginImageDims command

#### [SetPluginImageDims(wPixels,hPixels],option1], option2|,option3|,option4||||)]

Sets the size of images saved to disk with an image file export plugin using FileSaveAs. If either the width or height parameter is set to 0 then the corresponding document window dimension will be used, depending on the resolution setting (option3).

> Color depth. Valid values with all file export plugins distributed with DPlot are 8 (256 color), 24 (16 million colors), or 32-bits (PNG only). The JPEG plugin will ignore this entry (JPEGS are always 24bit). Likewise, GIF's are forced to 8-bits. If this option is omitted. DPlot will force it to the current bit depth of the display. 32-bit PNG images will be saved with an alpha channel, with the plot background color set to

transparent. For this image format DPlot will temporarily turn off the system's "Smooth screen fonts" and/or ClearType settings when saving an image, and also temporarily turn off antialiasing.

A combination of:

1

For 8-bit BMP files, used to specify whether the file should be runlength encoded (=1) or

#### option2

option1

#### ✓JR √Viewer

uncompressed (=0).

For 8-bit images, specifies whether a better (though much slower) color quantization method should be used to convert the input 16-, 24-, or 32-bit image to 8bit. The default color quantization method (=0) is very fast and works well for most images, but not well for background colors other than white. If your 8-bit images seem to be missing colors, use option2=2. NOTE: this flag is ignored for BMP files if the 1 bit is set; the BMP plugin contains a separate (and slower still) color quantization method for RLE bitmaps.

Resolution setting, in dots per inch (dpi). If a value less than 100 (including 0) is used then the saved image will essentially be a screenshot; the display's resolution is used. If greater than 100 then font sizes, line widths, the plot size and other dimensions will be scaled up by *resolution*/(100 dpi), so that the saved image will have the same proportions as the graph shown on your display. For best results with a resolution setting greater than 100, always specify the plot size with a <u>Size</u> command or the <u>Extents/Intervals/Size</u> menu command. Otherwise you risk generating an *Insufficient memory* error for high resolution settings.

#### JPEG

2

Quality setting, 1-5. 1 produces the smallest files, though the worst quality. 5 produces a lossless image and a much larger file than the other settings. The default is 4 ("good").

#### GIF

If non-zero, the plot background color will be set to transparent. For this image format DPlot will temporarily turn off the system's "Smooth screen fonts" and/or ClearType settings when saving an image, and also temporarily <u>turn off</u> <u>antialiasing</u>.

NOTE: For images with many unique colors and/or antialiased lines and symbols and for 3D views

option3

option4

of 3D/4D data in which one of the light source models is used, you will get better results saving the image as 24- rather than 8-bits (*option1*). If file size is paramount you can then use a dedicated image editor like <u>Adobe Photoshop</u> or <u>Paint Shop Pro</u> to reduce the color depth, with better results than you will get by saving to an 8-bit image in DPlot.

#### See also

**Image Properties** 

#### 5.2.204 SetWindowPos command

### [SetWindowPos(index,hwndInsertAfter,x,y,cx,cy,uflags JR Viewer

### )]

Similar to the Windows API SetWindowPos command.

#### index

- -1 document associated with the current DDE conversation.
- 0 main application window.
- 1-32 index of a child document window.

For *index*=0 (main application window), X and Y specify the window position in screen coordinates. For child document windows, X and Y specify the window position relative to the upper-left corner of the client area.

#### hWndInsertAfter

Handle to the window to precede the positioned window in the Z order. This value is ignored for all but *index*=0 (main application window). This parameter must be a window handle or one of the following values:

Value	Meaning
HWND_BOTTOM	Places the window at the bottom of the Z order. If the <i>hwndlnsertAfter</i>
(=1)	parameter identifies a topmost window, the window loses its topmost status
	and is placed at the bottom of all other windows.
HWND_NOTOPMO	Places the window above all non-topmost windows (that is, behind all topmost
ST (=-2)	windows). This flag has no effect if the window is already a non-topmost
	window.
HWND_TOP (=0)	Places the window at the top of the Z order.
HWND_TOPMOST	Places the window above all non-topmost windows. The window maintains its
(=-1)	topmost position even when it is deactivated.

#### X

Specifies the new position of the left side of the window, in client coordinates.

#### Y

Specifies the new position of the top of the window, in client coordinates.

#### СХ

Specifies the new width of the window, in pixels.

#### су

Specifies the new height of the window, in pixels.

#### uFlags

Specifies the window sizing and positioning flags. Most of these flags should only be used for the main application window. DPlot will re-activate the document associated with the current DDE conversation after processing this command, if necessary. This parameter can be a combination of the following values:

API Constant	Value	Meaning
SWP_ASYNCWINDOWP OS	0x4000 (16384)	If the calling thread does not own the window, the system posts the request to the thread that owns the window. This prevents the calling thread from blocking its execution while other threads process the request.
SWP_DEFERERASE	0x2000 (8192)	Prevents generation of the WM_SYNCPAINT message.
SWP_FRAMECHANGED	0x0020 (32)	Sends a WM_NCCALCSIZE message to the window, even if the window's size is not being changed. If this flag is not specified, WM_NCCALCSIZE is sent only when the window's size is being changed. Draws a frame (defined in the window's class description) around the window.
SWP_HIDEWINDOW	0x0080 (128)	Hides the window.
SWP_NOACTIVATE	0x0010 (16)	Does not activate the window. If this flag is not set, the window is activated and moved to the top of either the topmost or non-topmost group (depending on the setting of the <i>hWndInsertAfter</i> parameter).
SWP_NOCOPYBITS	0x0100 (256)	Discards the entire contents of the client area. If this flag is not specified, the valid contents of the client area are saved and copied back into the client area after the window is sized or repositioned.
SWP_NOMOVE	0x0002 (2)	Retains the current position (ignores the $X$ and $Y$ parameters).
SWP_NOOWNERZORDE R	0x0200 (512)	Does not change the owner window's position in the Z order.
SWP_NOREDRAW	0x0008 (8)	Does not redraw changes. If this flag is set, no repainting of any kind occurs. This applies to the client area, the nonclient area (including the title bar and scroll bars), and any part of the parent window uncovered as a result of the window being moved. When this flag is set, the application must explicitly invalidate or redraw any parts of the window and parent window that need redrawing.
SWP_NOSENDCHANGIN	0x0400	Prevents the window from

G	(1024)	receiving the WM_WINDOWPOSCHANGING message.
SWP_NOSIZE	0x0001 (1)	Retains the current size (ignores the <i>cx</i> and <i>cy</i> parameters).
SWP_NOZORDER	0x0004 (4)	Retains the current Z order (ignores the <i>hWndInsertAfter</i> parameter).
SWP_SHOWWINDOW	0x0040 (64)	Displays the window.

#### **Examples**

You can force DPlot's main window to stay on top of all other windows, retaining its current size and position with

SetWindowPos(0, -1, 0, 0, 0, 0, 3)

and turn off the stay-on-top state with SetWindowPos(0,-2,0,0,0,0,3)

These are equivalent to toggling the Keep DPlot on top menu item on the View menu.

# 5.2.205 Shell command

#### Shell(*command*,*nShowState*)] Macros Only ×JR √Viewer

Executes another program. *command* is the name of the program to execute along with any required arguments or command line switches. If the filename extension is omitted, ".EXE" is assumed. *nShowState* corresponds to the style of the window in which the program is to be executed. If omitted, defaults to 1.

The following table identifies the values for *nShowState* and the resulting style of window:

ValueWindow style

- 1 Normal with focus.
- 2 Minimized with focus.
- 3 Maximized with focus.
- 4 Normal without focus.
- 6 Minimized without focus.

**Note:** Macro processing continues as soon as the Shell function returns. You cannot depend on a program started with Shell to be finished executing before the macro commands following the Shell function in your macro are executed.

If the *command* includes spaces (e.g. c:\Program Files) **and** any commandline parameters, the executable name should be surrounded by double quotation marks, as in:

Shell(""c:\Program Files\My Application\test.bat" param1 param2",1)

### 5.2.206 ShowCaption command

#### [ShowCaption(state)]

#### ✓JR XViewer

Turns on (*state*=1) or off (*state*=0) the feature of drawing the document caption in the lower right corner of the window or page.

# 5.2.207 ShowCurve command

# [ShowCurve(index)]

Makes visible curves hidden by the <u>HideCurve</u> command. If *index* is less than or equal to 0, all curves in the document are made visible. Otherwise the *index'th* curve is made visible.

# 5.2.208 ShowDate command

# [ShowDate(date)]

Controls the display of the date in the upper right corner of the plot.

#### dateAction

- 0 Do not display the date
- 1 Display today's date
- >1 Convert *date* from Excel serial number to calendar date.

## 5.2.209 Size command

#### [Size(state,X,Y)]

For *state* = 0, DPlot automatically calculates the size of a plot such that it fills as much of the document window (or printed page or bitmap dimensions) as possible; X and Y are ignored. For *state*  $\neq$  0, X and Y specify the physical size of the plot axes, in inches.

# Examples: [Size(0)]





#### √JR XViewer

#### ✓JR XViewer

Over-well (2)	tern management	Carrier Car Are Area Street &	2012 - C
The last fact Options Generals into View Window Feet		STR III. A	
나는 바람 가 가 옷 다 눈 옷 가 가 다 만만 다 다 다		0 3 3 9 T	
	Bicycling	Caloric Expenditures	
1.700			
1,600	Rider weight, b	New to DPort? A good piece to	
1,600	- 190	per stated is the <u>perting</u> theted sector of the Help Te. Also take	
1.400	- 180	a look al the three do I strade the example gods? so loo	
	160		
1,300	- 150		
1,200	140		
	130		
1,100	110		
3 1,000			
ž			
3			
§ 800			
5 200			
600			
500			
470			
300		These curves can be approximated by kg_urCatmovBorV/25 - 1.85	
200		The values returned by this equation are valid for a relatively fait surface.	
		ne wind, and a standard road bite with	
100			
•	9 10 11 12 13 14	15 16 17 18 19 20 21 22 23 24 25	
	Av	erage opeed, we're	

#### NOTES:

In older versions with a non-zero state, there was a 4th WYSIWYG parameter that determined whether DPlot would adjust font sizes so that they were proportional to the size of the plot on all devices. This is set by default in more recent versions. If the specified width and height will not fit on the output (whether a display or a printout or a saved bitmap image), then font sizes are scaled down so that they have the same proportion to the size of the plot. For old macros and calls to DPlot Command, there is no danger in including a WYSIWYG 4th parameter, though it will be ignored.

If you want all relative sizes of fonts, line widths, symbol sizes, etc. to be the same when creating a plot from your own application, then using the Size command is highly recommended.

#### See also

Extents/Intervals/Size menu command

# 5.2.210 SplitDateTime command

#### [SplitDateTime(split)]

For **split** = 0, dates and times used with the Calendar or Date and Time number formats will be drawn on the same line. If **split** is non-zero, the time will be drawn below the date.

# 5.2.211 SquareLogCycles command

#### [SquareLogCycles(state)]

Forces the physical distance between log cycles to be the same on the X and Y axes for *state=*1.

# 5.2.212 StartStop command

#### [StartStop(start,stop)]

Sets the start and stop times for binary files. The StartStop command should be issued before a FileOpen command on a binary file.

#### ✓JR XViewer

# ✓JR XViewer

# 5.2.213 Stat AddData command

#### [Stat AddData(n,y1,y2,...,yn)]

✓JR XViewer

Adds ones or more amplitudes to the group selected with the SelectCurve command. If no group has been selected with SelectCurve, or if there is currently no group, adds data to the first group (which in the latter case will be unlabeled).

## 5.2.214 Stat AddGroup command

[Stat_AddGroup("name",n,y1,y2,...,yn)] ✓JR XViewer

Adds a new group with *n* amplitudes and group label *name*.

# 5.2.215 Stat Box command

#### [Stat Box(Draw,Color,Width,BorderWidth,BorderColor JR XViewer )1

Controls the appearance of the box in a box-and-whisker plot.

Draw 0 to **not** draw the box, 1 to draw the box

Color Color used to draw the box, red + green*256 + blue*65536

Width Width of the box, in 1000'th inch units

BorderWiWidth of the line used as a border for the box, in 1000'th inch units. Set to 0 for no dth border.

BorderC Color used for the border line around the box, red + green*256 + blue*65536 olor

## 5.2.216 Stat_BWOptions command

- [Stat BWOptions(MedianOrMean, ✓JR XViewer
  - DrawWhiskers.

StdDevorErr. BoxLimits. WhiskerLimits. OutlierLimits, ExtremeLimits)]

Specifies several drawing parameters for box-and-whisker plots. MedianOrMean =

0 for a median-based plot, 1 for a mean-based plot

#### DrawWhiskers =

0 if whiskers should not be drawn, 1 to draw whiskers

#### StdDevorErr =

0 if mean-based limits are multipliers for the group standard deviation, 1 if limits are multipliers for the group standard error (group standard deviation divided by the square root of the number of samples).

#### BoxLimits =

Extents of the box. For median-based plots, this value should be a percentage (0-100) of the range in amplitudes for the group. For mean-based plots, a multiplier for the group standard deviation or standard error (dependent on the StdDevorErr value).

#### WhiskerLimits =

Extents of the whiskers. For median-based plots, this value should be a percentage (0-100) of the range in amplitudes for the group. This value is not used for median-based plots unless the stat_whisker Option value (see below) is 0. For mean-based plots, a multiplier for the group standard deviation or standard error (dependent on the StdDevorErr value).

## OutlierLimits =

Limits defining outliers (distance from mean or median). For median-based plots, a multiplier for the interquartile range (box limits). For mean-based plots, a multiplier for the group standard deviation or standard error (dependent on the StdDevorErr value).

## ExtremeLimits =

Limits defining extremes (distance from mean or median). For median-based plots, a multiplier for the interguartile range (box limits). For mean-based plots, a multiplier for the group standard deviation or standard error (dependent on the StdDevorErr value).

# 5.2.217 Stat DotOptions command

# [Stat DotOptions(Grouping, Symbol, Size, Color, $\checkmark$ JR ×Viewer

Climits, Cstyle, Cwidth, Csymbol, Csize, Ccolor

)]

Dot graph options.

Groupin0 to plot individual data points scaled to their actual amplitude. Identical points are stacked. = 1 to group data points into bins, as in a histogram.

- Symbol Symbol style used for individual data points. See symbol codes under the description for the SymbolType command.
- Size Height of symbol, in 1000'th inch units
- Color used for individual data points, red + green*256 + blue*65536 Color
- *Climits* Limits of confidence limit line, expressed as a multiplier for the group standard error.
- Cstyle Line style used for confidence limit line.
  - 1=solid line 2=long dash 3=dotted 4=dash dot 5=medium dash 6=dash dot dot
- *Cwidth* Pen width used for confidence limit line, in 1000'th inch units.

Csymb Symbol style to draw at the ends of the confidence limits line. See symbol codes under the ol description for the <u>SymbolType</u> command.

Csize Height of CSymbol, in 1000'th inch units.

Ccolor Color used for confidence limits, red + green*256 + blue*65536

# 5.2.218 Stat DrawLegend command

# [Stat DrawLegend(draw)]

If *draw* is non-zero, a legend will be drawn with 1D statistics plots. If followed by 0, the legend is not drawn.

# 5.2.219 Stat GrandMean command

[Stat GrandMean(Flags,Width |,Color)]]

© 2014 HydeSoft Computing, LLC
- Flaces = If the 1 bit is set (Flaces is odd), draws the grand mean/median through the mean or median of all values in all groups. If the 2 bit is also set (Flags=3), the legend entry includes the value of the grand mean/median, as in "Grand median (56.2)".
- Width = width of line. in 1000'th inch units
- **Color** = Color used for drawing the grand mean/median, red + green*256 + blue*65536. If omitted, black is used.

#### 5.2.220 Stat_GroupInfo command

#### [Stat GroupInfo(Draw,Digits)]

Draw = Specifies which entries to draw in the upper left corner of each group. Combine the following values:

1 = mean (or median for median-based box-and-whisker plots)

- 2 = standard deviation (IQR for median-based box-and-whisker plots)
- 4 = coefficient of variation (%) (ignored for median-based plots)
- 8 = sample size

**Digits** = Number of significant figures to use for numbers.

#### 5.2.221 Stat LimitsAsCL command

#### [Stat_LimitsAsCL(state)]

If state is non-zero, then the legend for all limits on mean-based plots will be shown as a percent confidence value rather than a multiplier for the standard error. This command has no effect if limits are a function of standard deviation (Stat BWOptions StdDevorErr=0) rather than standard error.

#### 5.2.222 Stat MeanFormat command

#### [Stat_MeanFormat(SymbolOrLine,Style,Width, ✓JR XViewer Symbol,Size,Color)]

Controls the appearance of the mean or median on 1D statistics plots, SymbolorLine =

- 0 to draw the mean/median with a symbol
- 1 for a line segment
- Style = line style used if SymbolorLine=1 to drawn mean/median
  - 1=solid line 4=dash dot 2=long dash 5=medium dash
  - 3=dotted 6=dash dot dot
- Width = line width used, in 1000'th inch units
- Svmbol = symbol style to use if SymbolorLine=0. See symbol codes under the
- description for the <u>SymbolType</u> command.
- height of symbol, in 1000'th inch units Size =
- Color = Red + Green*256 + Blue*65536, where all components range from 0 to 255.

#### 5.2.223 Stat Outliers command

#### [Stat Outliers(Draw, Outlier Symbol, Outlier Color, ✓JR XViewer Extreme_Symbol, Extreme_Color)]

Controls the appearance of outliers on a box-and-whisker plot.

0 for no outliers or extremes, 1 to draw outliers, 2 to draw extremes but not outliers, 3 to Draw = draw outliers and extremes

#### Outlier Symbol =

Symbol style to use for outliers. See symbol codes under the description for the

#### JR XViewer

## SymbolType command

Outlier Color = Color used for outlier symbols, red + green*256 + blue*65536 Extreme Symbol = Symbol style to use for extremes (styles defined above). Extreme Color = Color used for extreme symbols, red + green*256 + blue*65536

#### 5.2.224 Stat_PlotType command

[Stat PlotType(box or dot)]

Draws a dot graph for a non-zero box_or_dot, otherwise draws a box-and-whisker plot.

#### 5.2.225 Stat RefLine command

#### [Stat RefLine(draw,value,style,width,color,"text")] ✓JR XViewer

Obsolete, but supported. Preferred command is Refline.

Adds a reference line to the plot, drawn across all groups. Each plot may have up to 2 reference lines (use Stat Refline twice to add the second line).

- **Draw** = 1 to draw the line. 0 if line should not be drawn.
- Value = Amplitude of the line
- Style = Line style to use for line 1=solid line 2=long dash 3=dotted 4=dash dot 5=medium dash 6=dash dot dot

Width = Width of pen used to draw the line, in 1000'th inch units

**Color** = Color of the line, red + green*256 + blue*65536

*Text* = Text that appears in the legend. Use "" for no legend entry.

#### 5.2.226 Stat Rotated command

#### [Stat Rotated(rotated)]

If *rotated* is non-zero, the plot is rotated such that groups are along the vertical axis and amplitudes on the horizontal axis.

#### 5.2.227 Stat Scaling command

#### [Stat_Scaling(scaling)]

By default the amplitude axis on 1D plots uses linear scaling. If scaling is non-zero, the amplitude axis will use a logarithmic scale. Non-positive extents or data values (if the extents are not forced to be positive) will result in the plot reverting to linear scaling.

#### 5.2.228 Stat Whiskers command

#### [Stat_Whiskers(Option,Style,Width,Color)]

Defines the appearance of whiskers on a box-and-whisker plot.

#### ✓JR XViewer

✓JR XViewer

✓JR XViewer

**Option** = Used only for median-based plots

0=whisker limits are expressed as a percentage 1=whiskers are drawn between non-outliers 2=whiskers include the entire data set.

 Style =
 Line style to use

 1=solid line

 2=long dash

 3=dotted

 4=dash dot

 5=medium dash

 6=dash dot dot

 Width =

 Width of pen used to draw whiskers, in 1000'th inch units.

 Color =

 Colors used for whiskers, red + green*256 + blue*65536

#### 5.2.229 SubtractCurves command

#### [SubtractCurves(Curve1,Curve2)]

Generates a new curve with amplitudes equal to the amplitudes of *Curve1* minus the amplitudes of *Curve1*.

XJR XViewer

X values for the new curve are taken from *Curve1*, subject to whether <u>FXY1Y2_XFrom</u> with a nonzero *state* value has been previously used. Linear interpolation is used to find the amplitude of *Curve2* at each X.

See also <u>Y=f(X,Y1,Y2,...)</u> menu command

#### 5.2.230 SwitchX command

#### [SwitchX(Curve/,options/)]

Uses the amplitudes of *Curve* as the independent variable (X values) of all other curves and shows the results in a new document. *Curve* must have monotonically increasing X values. If *options* is omitted, the new curves will be sorted in increasing order of the new independent variable and placed in a new document. If *options* is used and bit 1 is **not** set (e.g. **[SwitchX(***Curve***,0)]**, the curves will not be sorted. If bit 1 is set, the curves will be sorted on X. If the 2 bit of *options* is set, the existing curves will be replaced by the results of this operation. If bit 2 is **not** set, the result will be placed in a new document.

#### Example:

© 2014 HydeSoft Computing, LLC

mand

(,options))]

Subscriptions (X) Subscriptions (X) Subscriptions (X) Subscriptions (X) Subscription (X)





#### 5.2.231 SymbolClip command

#### [SymbolClip(state)]

#### XJR XViewer

For non-zero *state*, clips symbols at the boundaries of XY plots. For the default *state=*0, symbols within the plot boundaries are drawn without regard to clipping at the plot borders.

Example:

#### [SymbolClip(0)]



#### [SymbolClip(1)]





## 5.2.232 SymbolCycleColors command

#### [SymbolCycleColors(curve,state)]

If **state** is non-zero, symbols in **curve** cycle through the color palette rather than using the color associated with that **curve**. The length of the cycle is limited to the smaller of the number of curves on the plot and the size of the color palette (currently 16). If **curve** is set to -1, this command affects **all** curves.

See also

Symbol/Line Styles menu command

#### 5.2.233 SymbolFillBackground command

#### [SymbolFillBackground(state)]

# If *state* is non-zero, hollow symbols (not including Wingdings characters) will be filled with the plot background color, hiding any objects in the background including line segments and error bars for the same curve. If *state*=0, those symbols will be drawn with a transparent interior. The default plot background color is the window background color set with Control Panel. The plot background color may be changed with a BkColor command.

Example:



Oymbol/ Line Otyles mend command

#### 5.2.234 SymbolFrequency command

#### [SymbolFrequency(/curve,/frequency)]

✓JR XViewer

✓JR XViewer

Controls the frequency of symbols for curve number *curve* on XY plots and data set *curve* on 3D/4D scatter plots. If *frequency*=1, symbols are drawn at every data point, if=2, symbols are drawn at every other point, etc. If *curve* is set to -1 or omitted, the *frequency* value is applied to **all** curves/data sets.

See also Symbol/Line Styles menu command

#### 5.2.235 SymbolOpacity command

#### [SymbolOpacity(curve,opacity)]

Controls the opacity of symbols in an XY plot. Default opacity value is 100 (%). Valid values range

from 0 (completely transparent) to 100 (opaque). If *curve* is set to -1 or omitted, the *opacity* value is applied to **all** curves.

Symbol opacity < 100 is ignored for output to standard and enhanced metafiles. (This is because partially transparent symbols are drawn pixel-by-pixel, which would result in very large metafiles and in any case defeat the scalability of metafiles.) This setting is also ignored for displays with color depth of 256 or fewer colors.

The <u>SymbolOutline</u> setting is ignored for opacity values < 100: outlines of solid symbols will not be drawn; symbols will be drawn with a single color.

This feature might be useful for showing the density of data points. For example, here is an example using standard opacity=100% symbols:



and here is the same image with the symbol size increased from 0.10" to 0.30", opacity set to 5%:



Symbol/Line Styles menu command

#### 5.2.236 SymbolOutline command

#### [SymbolOutline(state)]

If *state* is non-zero, solid filled symbols will be outlined in black. If *state*=0, symbols are drawn with a single color.

See also <u>SymbolOutlineColor</u> <u>Symbol/Line Styles</u> menu command

#### 5.2.237 SymbolOutlineColor command

#### [SymbolOutlineColor(index,color)]

Sets the color for the *index'th* curve of an XY plot to the *color* (blue*256+green)*256+red, where the red, green, and blue components range from 0 to 255. *color* may use hexadecimal notation: 0x00FF80 represents red=128, green=255, blue=0. If *index* = -1, then *color* is used for ALL curves. This command automatically turns <u>SymbolOutline</u> on.

See also SymbolOutline ✓JR XViewer

Symbol/Line Styles menu command

#### 5.2.238 SymbolSize command

#### [SymbolSize(curve,size)]

Sets the height of the symbols used for curve number *curve* to *size* 1000'ths of an inch (or *size* Y units, depending on whether **SymbolSizeUnits** is used - see below). If *curve* is set to -1, this command changes the symbol size used by **all** curves.

For 3D/4D surface plots, *curve*=1 sets the size of the filled squares drawn at each data point if "data points" is checked on the <u>Contour Options</u> dialog, or equivalently the <u>SymbolType</u> is non-zero. The squares will always be an odd number (minimum=3) of pixels/dots, so that the square will be centered on the data point it represents.

See also

Symbol/Line Styles menu command

#### 5.2.239 SymbolSizeUnits command

#### [SymbolSizeUnits(state)]

Specifies the units for symbol sizes. If *state*=0 (default), symbol sizes are in 1000'ths of an inch. Symbols will be drawn in the same size when you zoom in/out or change the plot extents. If *state*=1, symbol sizes are proportional to the Y axis values (Z axis for 3D scatter plots). If you zoom in/out or change the plot extents, symbols are resized accordingly.

Example:

SymbolSizeUnits(0) SymbolSize(-1,50)

sets the size of all symbols to 0.05 inches, while:

SymbolSizeUnits(1)
SymbolSize(-1,50)

sets the size of all symbols to 50 Y units. If the Y axis extends from 0 to 1000, for example, then symbols will be roughly 1/20 of the plot height in this case. This setting is ignored for logarithmic or probability scales on the Y axis. If used with these scale types, it will be automatically reset to 0 and all symbol sizes will be reset to a default height (approximately 1/40 of the plot height).

#### See also

Symbol/Line Styles menu command

#### 5.2.240 SymbolType command

#### [SymbolType(curve,type)]

Sets the symbol type. Set *type* to 0 for no symbols. Valid *type* values are shown below. If *curve* is set to -1, this command changes the symbol style used by **all** curves/data sets.

## ✓JR XViewer

✓JR XViewer



Symbols 28 and 284 are identical to symbols 9 and 265, respectively, with the exception of vertical alignment. Symbols 9 and 265 are placed such that the geometric center of the triangle is at the data point. Symbols 28 and 284 are vertically aligned with symbols 7 and 263.

Symbol type 389 is only available in 3D scatter plots when "Symbol size in Z units" (see <u>SymbolSizeUnits</u>) is selected. In that case, this symbol (unlike all others) will be drawn using the perspective settings for the plot. To control perspective programmatically, use the <u>ContourView</u> command.

**Example:** 



You can also specify a character from the Wingdings, Wingdings 2, or Wingdings 3 fonts to use as a symbol. Note: These fonts must be installed on the end user's system for this feature to work properly. In this case *type* is the character code plus 512 (0x0200) for Wingdings, 768 (0x0300) for Wingdings 2, or 1024 (0x0400) for Wingdings 3. For example *type* = 0x0393 (decimal 915) specifies character 0x93 (147) from the Wingdings 2 font. Character codes should be greater than 32 (space) and must be less than or equal to 255 (0xFF).

You can view any of the Wingdings fonts using the Character Map applet that is installed with Windows (generally under Programs>Accessories), or by selecting the <u>Symbol/Line Styles</u> command on the Options menu and clicking the *Other...* button.

For 3D/4D surface plots, setting the symbol type to a non-zero value is equivalent to checking the

"Data points" box on the Contour Options dialog. The actual type is ignored in this case; symbols will always be filled black squares. You can control the size of the squares with the <u>SymbolSize</u> command.

#### See also

Symbol/Line Styles menu command

#### 5.2.241 TextFont command

#### 

Sets various parameters for one of the six fonts used by DPlot. Index should be one of the following:

- 1 Numbers along the axes
- 2 First title line
- 3 Second and third title lines
- 4 X axis label
- 5 Y axis label(s)
- 6 Legend
- 7 Note(s)
- 8 Z axis label

**Points** is the size of the font, where 1 point = 1/72 inch. **Weight** is a number between 100 for "thin" and 900 for "heavy" or 0 for "don't care", in which case Windows picks the font weight. Normal text has a weight of 400, bold text has a weight of 700. Not all font weights are available for all possible fonts... Windows selects the closest weight available. **Italic** is 1 for italics, 0 otherwise. **R**,**G**,**B** are the red, green, and blue components used in the font color, ranging from 0 to 255 (**R**,**G**,**B** = 0,0,0 corresponds to black). **"Face"** is the font's face name,e.g. "Arial" or "Times New Roman".

All parameters other than *Index* are optional. For example, the following sets the color of the first title line to red and leaves all other parameters at their current values:

[TextFont(2,,,,255,0,0)]

#### 5.2.242 TextNote command

#### [TextNote("Note",X,Y,Frame,Opaque,Button,TabStops)] JR Viewer

Obsolete (though still supported). Use TextNoteEx.

Adds a notation to the plot. **Note** is the text of the notation. **X**, **Y** are the coordinates of the upper left corner of the note, expressed as a percentage of the plot size (0.0 to 1.0). **Frame** = 1 to draw a rectangle around the note. **Opaque** = 1 to hide any objects intersecting the note. **Button** = 1 if the note is "activated" with a button. **TabStops** is the number of character cells used for tab positioning. See the <u>Add/Edit Note</u> command for more information.

To break a note into multiple lines, you may either embed carriage return-line feed sequences into the note text, or include the literal sequence "\n". The latter method is the only way to include line breaks in a note produced by a macro. Examples:

In VB/VBA:

```
Control.LinkExecute "[TextNote(""Line 1" & chr$(13) & chr$(10) & "Line
2"",0.1,0.1,1,1,0,8)]"
```

In C/C++, using DPLOTLIB:

#### DPlot_Command(DocNum,

"[TextNote(\"Line 1\r\nLine 2\",0.1,0.1,1,1,0,8)]");

#### In Fortran, using DPLOTLIB:

ret=DPlot_Command(DocNum,

'[TextNote("Line 1\nLine 2",0.1,0.1,1,1,0,8)]'//char(0))

In a macro:

```
[TextNote("Line 1\nLine 2",0.1,0.1,1,1,0,8)]
```

#### See also

Add/Edit Note menu command DeleteNote macro command

#### 5.2.243 TextNote3D command

#### 

Adds a notation to the plot. This command is identical to <u>TextNoteEx</u>, with the addition of two additional Z coordinate parameters.

*Note* is the text of the notation, limited to 511 characters.

*X*, *Y*, *Z* are the coordinates of the alignment point of the note, expressed as a percentage of the plot size (0.0 to 1.0) or in data coordinates, dependent on the N_DATASPACE bit of the *Flags* setting. The *Z* parameter is meaningful only for 3D views of 3D/4D data with the N_DATASPACE bit set.

*Flags* is a combination of the following:

N_FRAME	0x0002	Draw a rectangle around the
N_OPAQUE	0x0004	Hide objects intersecting the note. If not set, note background
N_BUTTON N_DATASPACE	0x0008 0x0010	Note is "activated" by a button. X and Y are expressed in data space rather than the default ratios of plot size. If set, the note will remain in a fixed position relative to the plot's data values when you zoom or otherwise change plot extents. NOTE: For 3D or 4D plots, TextNote3D is preferred. TextNote3D provides a Z parameter for the note location. If TextNoteEx is used on a 3D or 4D plot with this bit set, the Z value is set to 0
N_HIDE	0x0020	The note will be hidden (not drawn). Visibility may be restored only by selecting the Add/Edit Note command on the Text menu
N_ARROW	0x0040	Draw a line from the note to the point <i>Xarrow</i> , <i>Yarrow, Zarrow</i> . Line width and arrowhead dimensions may be controlled with the <u>TextNoteLeader</u> command. <i>Xarrow, Yarrow,</i> and <i>Zarrow</i> are data space coordinates, always proportional

to the primary axes regardless of whether multiple axes are used or not. *Xarrow, Yarrow,* and *Zarrow* are ignored if this bit is not set. Vertical orientation. Default is horizontal. This bit is ignored for N_BUTTON notes.

*Alignment.* Default alignment point is the top left corner of the note. *Alignment* can be a combination of the following:

0x0080

- 0x0001 The reference point will be aligned horizontally with the center of the bounding rectangle.
- 0x0002 The reference point will be on the right edge of the bounding rectangle.
- 0x0004 The reference point will be aligned vertically with the middle of the bounding rectangle.
- 0x0008 The reference point will be on the bottom edge of the bounding rectangle.

*TabStops* is the number of character cells used for tab positioning. See the <u>Add/Edit Note</u> command for more information.

To break a note into multiple lines, you may either embed carriage return-line feed sequences into the note text, or include the literal sequence "\n". The latter method is the only way to include line breaks in a note produced by a macro. Examples:

In VB/VBA:

N VERT

```
Control.LinkExecute "[TextNote3D(""Line 1" & chr$(13) & chr$(10) & "Line 2"",0.1,0.1,0.5,0,0,8)]"
```

In C/C++, using <u>DPLOTLIB</u>:

DPlot_Command(DocNum,

"[TextNote3D(\"Line 1\r\nLine 2\",0.1,0.1,0.5,0,0,8)]");

In Fortran, using DPLOTLIB:

```
ret=DPlot_Command(DocNum,
```

```
'[TextNote3D("Line 1\nLine 2",0.1,0.1,0.5,0,0,8)]'//char(0))
```

In a macro:

[TextNote3D("Line 1\nLine 2",0.1,0.1,0.5,0,0,8)]

**bkColor** is the background color of the notation. If omitted or set to -1, the note background will be the same color as the plot background (set with <u>BkColor</u>). This value is ignored if the N_OPAQUE flag is not set.

#### See also

Add/Edit Note menu command DeleteNote macro command

#### 5.2.244 TextNoteEx command

Adds a notation to the plot.

*Note* is the text of the notation, limited to 511 characters.

*X*, *Y* are the coordinates of the alignment point of the note, expressed as a percentage of the plot size (0.0 to 1.0) or in data coordinates, dependent on the N_DATASPACE bit of the *Flags* setting.

Flags is a combination of the following: N_FRAME 0x0002

Draw a rectangle around the note.

N_OPAQUE	0x0004	Hide objects intersecting the note. If not set, note background
N_BUTTON N_DATASPACE	0x0008 0x0010	Note is "activated" by a button. X and Y are expressed in data space rather than the default ratios of plot size. If set, the note will remain in a fixed position relative to the plot's data values when you zoom or otherwise change plot extents. NOTE: For 3D or 4D plots, TextNote3D is preferred. <u>TextNote3D</u> provides a Z parameter for the note location. If TextNoteEx is used on a 3D or 4D plot with this bit set, the Z value is set to 0.
N_HIDE	0x0020	The note will be hidden (not drawn). Visibility may be restored only by selecting the Add/Edit Note command on the Text menu
N_ARROW	0x0040	Draw a line from the note to the point <i>Xarrow</i> , <i>Yarrow</i> . Line width and arrowhead dimensions may be controlled with the <u>TextNoteLeader</u> command. <i>Xarrow</i> and <i>Yarrow</i> are data space coordinates, always proportional to the primary X and Y axis regardless of whether multiple axes are used or not. <i>Xarrow</i> and <i>Yarrow</i> are ignored if this bit is not set
N_VERT	0x0080	Vertical orientation. Default is horizontal. This bit is ignored for N_BUTTON notes.

*Alignment.* Default alignment point is the top left corner of the note. *Alignment* can be a combination of the following:

0x0001 The reference point will be aligned horizontally with the center of the bounding rectangle.

0x0002 The reference point will be on the right edge of the bounding rectangle.

0x0004 The reference point will be aligned vertically with the middle of the bounding rectangle.

0x0008 The reference point will be on the bottom edge of the bounding rectangle.

*TabStops* is the number of character cells used for tab positioning. See the <u>Add/Edit Note</u> command for more information.

To break a note into multiple lines, you may either embed carriage return-line feed sequences into the note text, or include the literal sequence "\n". The latter method is the only way to include line breaks in a note produced by a macro. Examples:

In VB/VBA:

```
Control.LinkExecute "[TextNoteEx(""Line 1" & chr$(13) & chr$(10) &
    "Line 2"",0.1,0.1,0,0,8)]"
```

In C/C++, using DPLOTLIB: DPlot_Command(DocNum, "[TextNoteEx(\"Line 1\r\nLine 2\",0.1,0.1,0,0,8)]");

In Fortran, using DPLOTLIB:

ret=DPlot_Command(DocNum,

'[TextNoteEx("Line 1\nLine 2",0.1,0.1,0,0,8)]'//char(0))

In a macro:

[TextNoteEx("Line 1\nLine 2",0.1,0.1,0,0,8)]

**bkColor** is the background color of the notation. If omitted or set to -1, the note background will be the same color as the plot background (set with <u>BkColor</u>). This value is ignored if the N_OPAQUE flag is not set.

```
See also
```

Add/Edit Note menu command DeleteNote macro command

#### 5.2.245 TextNoteLeader command

#### [TextNoteLeader(ArrowH,ArrowW,LineWidth)]

Specifies the arrowhead height and width and line width used for lines drawn from text notations to referenced X,Y|,Z| coordinates when the N_ARROW bit of the *Flags* argument to a <u>TextNoteEx</u> or <u>TextNote3D</u> command is set. All dimensions are in 1000ths of inches. If *ArrowH* is omitted or set to 0, the default height of 0.15 inches is used. If you do not want an arrowhead, set *ArrowH* to -1. If *ArrowW* is omitted or set to 0, one half the arrow height will be used.

See also

Add/Edit Note menu command

#### 5.2.246 TextPointLabel command

## [TextPointLabel(*Where,X,Y,Frame,Opaque,|Options,* **VIR Viewer** *bkColor,|"Label"*)]

Attaches a label to a specific data point in the curve or data group most recently selected with the <u>SelectCurve</u> command. All parameters must be separated by commas.

- Options = Please note: The pipe symbols ( || ) surrounding Options in the above description indicate that this parameter is optional. Do not include these symbols with this parameter. Options is a combination of the following values. Note that bits 1,2, and 4 are mutually exclusive, as are 8 and 32.
  - 1 *Where* specifies the X coordinate of the point you want to label. The data point with X value closest to *Where* will be labeled. (This option is not valid for 1D plots.)
  - 2 Label the point with the maximum Y value. Where is ignored.
  - 4 Label the point with the minimum Y value. Again, *Where* is ignored.
  - 8 **X** and **Y** specify relative offsets from the labeled data point in plot coordinate space, rather than an absolute location. (This option is not valid for 1D plots.)
  - 16 Rotate label 90 degrees counterclockwise.
  - 32 **X** and **Y** specify relative offsets from the labeled data point in inches. Note that if you subsequently use the mouse to move the label that this option is automatically turned off.
  - 64 The label is mouse-activated. It will only be drawn when the user moves the mouse cursor over the point referenced by the label.

128 Label is center-justified horizontally. Default is left-justified.

- 256 Label is right-justified.
- 512 Label is top-justified vertically. Default is vertically centered.
- 1024 Label is bottom-justified.

If **Options** is omitted or does not have any of the 1, 2, or 4 bits set, then **Where** specifies the data point to label (1 through number of points in the curve or group).

If *Where*=-1 (and, again, none of *Options* bits 1,2, or 4 are set), then **all** points in the currently selected curve (**all** curves for SelectCurve(-1)) will be labeled.

- X,Y = Label coordinates in plot space if neither the 8 or 32 bits of Options is set.
   Offset from labeled point in plot space if the 8 bit of Options is set.
   Offset from labeled point in inches if the 32 bit of Options is set.
- *Frame* = If = 1, a rectangle will be drawn around the label.
- **Opaque** = If = 1, the label will hide any objects that intersect the label. If = 0, objects underneath the label will be visible.
- *bkColor* is the background color of the label. If omitted or if *Opaque* is 0, the background will be the same color as the plot background (set with <u>BkColor</u>).
- "Label" Text to use for the label. See the Help topic on the the <u>Label Points</u> command for more information. If "Label" has 0 length (...,Opaque,/Options,/"")]), then the amplitude (Y value) of the point will be used as the label. This is equivalent to checking "Use amplitude only" on the **Point Labels** dialog.

#### See also

Label Points menu command

#### 5.2.247 TextPointLabel3D command

#### [TextPointLabel3D(set,Xd,Yd,Xl,Yl,Frame, Opaque,/Options,bkColor,/"Label")]

✓JR XViewer

Attaches a label to a specific data point in a 3D surface. **NOTE:** Labels are only drawn in 2D views of the surface, not in 3D views. All parameters must be separated by commas.

#### Set Currently not used.

- Options = Please note: The pipe symbols ( || ) surrounding Options in the above description indicate that this parameter is optional. Do not include these symbols with this parameter. Options is a combination of the following values.
  - 16 Rotate label 90 degrees counterclockwise.
  - 32 **X** and **Y** specify relative offsets from the labeled data point in inches. Note that if you subsequently use the mouse to move the label that this option is automatically turned off.
  - 64 The label is mouse-activated. It will only be drawn when the user moves the mouse cursor over the point referenced by the label.
  - 128 Label is center-justified horizontally. Default is left-justified.
  - 256 Label is right-justified.
  - 512 Label is top-justified vertically. Default is vertically centered.
  - 1024 Label is bottom-justified.
- *Xd*, *Yd* = X and Y coordinates of the data point to label. DPlot will find the data point closest to this location.
- *XI*, *YI* = Label coordinates in plot space if the 32 bit of *Options* is not set.

Offset from labeled point in inches if the 32 bit of **Options** is set.

- *Frame* = If = 1, a rectangle will be drawn around the label.
- **Opaque** = If = 1, the label will hide any objects that intersect the label. If = 0, objects underneath the label will be visible.

- *bkColor* is the background color of the label. If omitted or if *Opaque* is 0, the background will be the same color as the plot background (set with BkColor).
- "Label" Text to use for the label. See the Help topic on the the <u>Label Points</u> command for more information. If "Label" has 0 length (...,Opaque,/Options,/"")]), then the amplitude (Y value) of the point will be used as the label. This is equivalent to checking "Use amplitude only" on the **Point Labels** dialog.

#### See also

Label Points menu command

#### 5.2.248 TextPointLabelLeader command

#### 

Specifies the arrowhead width and height and the line used in point labels, in 1000'ths of inches. The default arrowhead size is 150 (0.15 inches) tall by 75 wide. The default line width is 0 (1 pixel/ printer dot). Default values will be used for 0 values. If the arrowhead height is set to a negative number, arrowheads will not be drawn.

#### See also

Label Points menu command

#### 5.2.249 TickInterval command

#### [TickInterval(state,dx,dy,dz)]

#### ✓JR XViewer

If **state** is non-zero, specifies the tick mark interval to use on the X, Y, and Z axes (3D plots only). If **state**=2 and the current document contains a polar plot, then **dx** is in degrees rather than the default radians. If **state**=0, DPlot automatically selects a tick mark interval.

All parameters must be separated by commas. *dx*, *dy*, and *dz* are optional. Omitted parameters are represented by 2 successive commas with no intervening spaces. If omitted then the current tick mark interval for the respective axis will be maintained. For example, to set the tick mark interval on the Y axis to 5 and leave the X (and Z for 3D plots) interval alone, use

#### [TickInterval(1,,5)]

To set only the X tick mark interval, the following commands are equivalent:

#### [TickInterval(1,5)] [TickInterval(1,5,)] [TickInterval(1,5,,)]

Alternatively, see the NumTicks command.

You can use equations for the tick interval on any axis. This might be handy if you do not know the extents of the data, but **do** know that you want, for example, the tick mark interval to be a power of 10:

TickInterval(1,=10.^(FLOOR(LOG10(\$XRANGE)-1)))

sets the tick mark interval on the X axis to an integer power of 10 such that there will be at least 10 intervals but less than 100. For more information see the list of <u>available functions</u> and the <u>text</u> <u>placeholders</u> topics.

#### See also

Extents/Intervals/Size menu command

#### 5.2.250 TickMarks command

#### [TickMarks(placement,major,minor)]

Specifies the location and size of tick marks. *Placement* is either 1 to draw tick marks inside the box surrounding the graph, 2 to draw tick marks outside the graph, or 3 for both inside and outside. Major and minor specify the length of major and minor tick marks in 1000ths of inches. All parameters are optional. The default values when you first install DPlot are 1,80,40.

#### 5.2.251 TimeFormat command

#### [TimeFormat("format")]

The format string controls the appearance of numbers using the Date and time or Time number formats. This is the same time format string that will appear in the *Time Format* box of the Number Formats dialog.

Time format notation: h = hour, m=minute, s=second, t=AM or PM (ignored for 24 hour clock) h = 12 hour clock, H = 24 hour clock hh or HH = leading zero on hour h or H = no leading zero on hour

#### 5.2.252 Title1 command

#### [Title1("string")]

Sets the first title line to string. Limited to 200 characters. For formatting codes (bold, italic, subscripts, color, etc.) see the description of the Title/Axes menu command.

See also

Title/Axes menu command

#### 5.2.253 Title2 command

#### [Title2("string")]

Sets the second title line to *string*. Limited to 200 characters. For formatting codes (**bold**, *italic*, subscripts, color, etc.) see the description of the Title/Axes menu command.

See also

Title/Axes menu command

#### 5.2.254 Title3 command

#### [Title3("string")]

#### JR XViewer

Sets the third title line to string. Limited to 200 characters. For formatting codes (bold, italic, subscripts, color, etc.) see the description of the Title/Axes menu command.

#### See also

## © 2014 HydeSoft Computing, LLC

✓JR XViewer

✓JR XViewer

#### ✓JR XViewer

#### 5.2.255 Title4 command

#### [Title4("string")]

#### ✓JR XViewer

Sets the fourth title line to *string*. Limited to 200 characters. For formatting codes (**bold**, *italic*, subscripts, color, etc.) see the description of the <u>Title/Axes</u> menu command.

See also

Title/Axes menu command

#### 5.2.256 TriangleFlags command

#### [TriangleFlags(flags)]

Changes the appearance of triangle plots. *Flags* is a combination of: 0x0001 Draw USDA soil classification lines and labels.

- 0x0002 Clockwise orientation (numbers increase in a clockwise direction around the plot rather than counterclockwise, and the Y axis is on the right side rather than the left). This orientation is normally used with USDA textural plots.
- 0x0008 Draw IUGS mafic rock classification lines and labels.
- 0x0010 Draw IUGS ultramafic rock classification lines and labels.
- 0x0020 Draw Ab-An-Or (Albite-Anorthosite-Orthoclase) mineral classification lines and labels.
- 0x1000 Range of each component is 0-1 rather than the default 0-100, and the total of X+Y+Z is 1.0 rather than 100.
- 0x0001, 0x0008, 0x0010, and 0x0020 are mutually exclusive they cannot be combined, in other words.

For more information see the Triangle Plot Options topic.

#### 5.2.257 TripartiteFlags command

#### [TripartiteFlags(flags)]

Changes the appearance of tripartite plots. *Flags* is a combination of:

0x0001 Draw intermediate acceleration and displacement lines (2,3,4,6,7,8, and 9 times a power of 10) and label the 5 times a power of 10 lines. By default, grid lines are only drawn at 1 and 5 times a power of 10, and only the 1 times a power of 10 lines are labeled.

#### See also

<u>Scale</u> macro command <u>Tripartite Options</u> menu command

#### 5.2.258 UseNameAsLegend command

#### [UseNameAsLegend(state)]

If state is non-zero, FileOpen and FileAppend operations on files consisting of 1 curve only will use

#### √JR ×Viewer

#### √JR XViewer

the filename as the legend of the curve.

#### 5.2.259 Vector Plot command

#### [VectorPlot(*curve*, AngleSource, MagnitudeSource, JR XViewer ColorLo,ColorHi,SizeLo,SizeHi|,"Heading"|)]

Sets vector plot options for the curve with 1-based index *curve*. Angular values are retrieved from the *AngleSource*'th curve; magnitudes from the *MagnitudeSource*'th curve. If either of these entries is non-positive, the vector plot option is turned off for *curve*. The *AngleSource* and *MagnitudeSource* curves will not be drawn (assuming both entries are positive).

By default, **ColorLo** and **ColorHi** are colors mapped to the low and high absolute values of the **MagnitudeSource** amplitudes. This behavior may be changed with a subsequent <u>VectorPlotOptions</u> (set 0x0040 bit) and <u>VectorPlotColors</u> command. To use the same color for all vectors, set **ColorLo** and **ColorHi** to the same values. Colors are in the form blue*65536+green*256+red, where blue, green, and red are between 0 and 255, inclusive. Colors may be specified in hexadecimal notation with a '0x' prefix. For example 0x00FFFF00 is cyan (blue=255, green=255, red=0). This obviously limits the number of colors in a vector plot to 2 (other than the blending between those colors). You can specify up to 8 colors with a subsequent <u>VectorPlotColors</u> command.

*SizeLo* and *SizeHi* are the lengths, in thousandths of inches, of the vectors corresponding to the low and high absolute values of the *MagnitudeSource* amplitudes or to the low and high magnitudes set with <u>VectorPlotColors</u>.

The optional "*Heading*" entry will be used as the heading for the color key, but only if the 0x0020 bit of the *flags* argument to a subsequent <u>VectorPlotOptions</u> command is set.

#### See also

Vector Plot menu command Vector Plot Colors macro command Vector Plot Options macro command

#### 5.2.260 VectorPlotColors command

#### [VectorPlotColors(*curve*,*N*,*Color1*,*Color2*,...,*ColorN*,, *JR* Viewer Magnitude1,Magnitude2,...,MagnitudeN)]

Sets up to 8 colors and (optionally) 8 magnitudes associated with each color for the vector plot associated with the 1-based *curve*. *N* is the number of following colors and optional magnitudes and must be between 2 and 8. Magnitude values are ignored unless you set the 0x0040 bit of *flags* with the <u>VectorPlotOptions</u> macro command. If magnitudes **are** included, this command **must** follow both VectorPlot and VectorPlotOptions with the 0x0040 bit set.

Colors are in the form blue*65536+green*256+red, where blue, green, and red are between 0 and 255, inclusive. Colors may be specified in hexadecimal notation with a '0x' prefix. For example 0x00FFFF00 is cyan (blue=255, green=255, red=0).

See also

Vector Plot menu command Vector Plot macro command Vector Plot Options macro command

#### 5.2.261 VectorPlotOptions command

[VectorPlotOptions(flags,arrowheight,linewidth)

	√JR	×Viewer
--	-----	---------

Sets several options for the appearance of vector plots.

*flags* is a combination of:

- 0x0001 Angles are specified in degrees rather than the default radians.
- 0x0002 Angles are measured clockwise. Default is counterclockwise.
- 0x0004 0 degrees is in the positive Y direction. Default is in the positive X direction.
- 0x0008 Vector centered on point. Default is with vector tail at the point.
- 0x0010 Discrete colors. Default is to blend colors between various magnitudes.
- 0x0020 Draw color key
- 0x0040 Specify magnitudes associated with each color. The default is to use equal intervals between the low and high values of the magnitude source curve.

Combine *flags* options by adding them.

arrowheight is the height of the vector arrowheads, in thousandths of inches. Default value is 150.

*linewidth* is the width of the vector lines, in thousandths of inches. A value of 0 results in 1 unit (1 pixel, 1 printer dot, etc.) lines.

#### See also

Vector Plot menu command Vector Plot macro command Vector Plot Colors macro command

#### 5.2.262 ViewRedraw command

#### [ViewRedraw()]

#### ✓JR √Viewer

✓JR √Viewer

✓JR √Viewer

Forces DPlot to redraw the plot. For real-time applications, DPlot performs several optimizations that may leave undesirable artifacts on a plot. For example, line segments may intersect the legend and/ or other plot features that were intended to be opaque. **ViewRedraw** causes DPlot to draw the plot correctly.

#### 5.2.263 ViewStatusBar command

[ViewStatusBar(visibility)]

Shows (visibility=1) or hides (visibility=0) the status bar.

#### 5.2.264 ViewToolBar command

#### [ViewToolBar(visibility,largebuttons)]

Shows (*visibility*=1) or hides (*visibility*=0) the toolbar, with 24x22 pixel (*largebuttons*=0) or 32x30 pixel (*largebuttons*=1) buttons.

#### 5.2.265 WindowCascade command

#### [WindowCascade()]

Overlaps all DPlot document windows, with the active document on top. Equivalent to the Cascade command on the Window menu.

#### 5.2.266 Window Tile command

[WindowTile()] or

#### [WindowTile(arrange)]

Displays all DPlot documents in windows that do not overlap. If arrange is omitted or equal to 0, document windows are arranged such that they are tall rather than wide (equivalent to the Tile command on the Window menu). If arrange is equal to 1, document windows are arranged such that they are wide rather than tall (equivalent to the Tile Horizontally command on the Window menu).

#### 5.2.267 XAxisLabel command

#### [XAxisLabel("string")]

Sets the X axis label to string. Limited to 200 characters. For formatting codes (bold, italic, subscripts, color, etc.) see the description of the Title/Axes menu command.

See also

Title/Axes menu command

#### 5.2.268 X2AxisLabel command

#### [X2AxisLabel("string")]

Sets the label for the second X axis to string. Limited to 200 characters. For formatting codes (bold, italic, subscripts, color, etc.) see the description of the Title/Axes menu command. This command will not result in any changes to the plot unless multiple X axes are used (though string will be preserved in that case and used whenever multiple X axes are used). The multiple axis setting is controlled either by the MultipleXAxes macro command or the Multiple Axes command on the Options menu.

See also

Title/Axes menu command

#### 5.2.269 XLabel command

#### [XLabel(X,"label")]

#### ✓JR XViewer

Obsolete, though supported. XYLabel is preferred. Draws the text **label** at the location X (in plot space) below the X axis.

#### 5.2.270 XY command

#### [XY(x,y)]

#### ✓JR XViewer

Adds a data point to the curve selected with the SelectCurve command. If this operation would cause the number of points in the curve to exceed the array sizes (set with FileArrays), DPlot returns DDE FNOTPROCESSED to a DDE client, or causes an error in the DPlot macro facility. The x and y value **must** be separated by a comma. Both the x and y values can be represented as times (e.g. 09:24:45.01 PM), dates, or date and time groups (with a space between the date and time).

✓JR XViewer

✓JR XViewer

**NOTE:** Executing multiple [XY()] commands on a Windows 95 or 98 system is very slow, while on Windows 2000 and later systems is only slightly slower than sending binary data via an XTYP_POKE transaction (as in the <u>DPLOTLIB</u> demo programs). If you are developing an application for users who might be using Windows 95 or 98, **and** yours is not a real-time application that plots results as they occur, you should consider either:

Using the [XYXY()] command to send multiple data pairs rather than individual [XY()] commands. Or:

Preface a series of [XY()] commands with a [PostponeRedraw()], and terminate the series of [XY()] commands with a [ViewRedraw()].

The former (using [XYXY()] will in all cases be faster but might not always be practical. The latter will delay redrawing the plot until the ViewRedraw command is sent.

#### See also

DeferMinMaxCheck macro command

XYXY macro command

#### 5.2.271 XYLabel command

#### [XYLabel(XY,location,offset,"label")]

Draws the text label at or near the X, Y, or Z axis, depending on the location and offset settings.

XY X or Y value, in plot space, where the label will be located.

*location* Specifies the axis near which the label will be drawn.

- 0 Lower X axis
- 1 Upper X axis
- 2 Left Y axis
- 3 Right Y axis
- 4 Zaxis

offset Distance, in thousandths of inches, between the axis and the label.

For 3D views of 3D or 4D data, the labels will always be drawn on the same side of the plot that the numbers along the coordinate axes are drawn.

This command is useful for replacing numeric values along an axis with non-numeric labels, particularly for bar charts. To hide the normal numbers appearing on an axis, set the number format to "None": [NumberFormat(0,12)] for the X axis, [NumberFormat(1,12)] for the Y axis.

#### 5.2.272 XYXY command

#### [XYXY(*n*,*x*1,*y*1,*x*2,*y*2,...,*xn*,*yn*)]

Adds *n* data points to the curve selected with the <u>SelectCurve</u> command. If this operation would cause the number of points in the curve to exceed the array sizes (set with <u>FileArrays</u>), DPlot returns DDE_FNOTPROCESSED to a DDE client, or causes an error in the DPlot macro facility.

This method of sending data to DPlot is considerably faster than the [XY()] command and is particularly useful in environments like Visual Basic that do not readily allow you to send binary data via DDE. Of particular importance, the maximum command string length that DPlot will process is 32768 characters, including the command itself and all delimiters and spaces. For this reason you should limit the number of points sent with XYXY to a maximum of about 1000. If your application requires sending more than 1000 points, use multiple XYXY commands in separate XTYP_EXECUTE (LinkExecute in VB jargon) transactions.

## √JR ×Viewer

#### See also

DeferMinMaxCheck macro command

#### 5.2.273 XYZ command

#### [XYZ(*n*,*x*1,*y*1,*z*1,*x*2,*y*2,*z*2,...,*xn*,*yn*,*zn*)]

✓JR ×Viewer

Adds *n* x,y,z triplets to a contour plot.

The maximum command string length that DPlot will process is 32768 characters, including the command itself and all delimiters and spaces. For this reason you should limit the number of points sent with XYZ to a maximum of about 1000.

**NOTE:** The maximum command string length for all tested versions of Microsoft Excel with DDEExecute is only 256 characters, which presents a few problems for this command. As an alternative, see the XYZEx command, which allows you to instruct DPlot as to whether the data should be triangulated and plotted or not.

You should **not** send a long series of XYZ commands with individual points. Instead, group the points into one XYZ command or, if that is not possible, use XYZEx to postpone generating a mesh until all points are sent.

If an error in the triangulation occurs (for example if all points are collinear) an error message is NOT generated. DPlot in this case assumes that one or more XYZ commands which would correct the situation will follow the current command.

See also

DeferMinMaxCheck macro command

#### 5.2.274 XYZ_To_XY command

#### [XYZ_To_XY(method,parameters....)]

Creates an XY plot from the intersection of a vertical plane with the active 3D surface plot. This command is equivalent to the <u>XY Plot from 3D Surface</u> menu command.

The first parameter (*method*) gives the method for specifying the intersection plane and must be 0, 1, or 2. For a line Y=mX+b use

#### [XYZ_TO_XY(0,m,b)]

For a constant X plane, use [XYZ_TO_XY(*1,x*)] and to define the plane with two points (x1,y1) and (x2,y2) use [XYZ_TO_XY(*2,x1,y1,x2,y2*)]

#### 5.2.275 XYZEx command

#### [XYZEx(replot,n,x1,y1,z1,x2,y2,z2,...,xn,yn,zn)]

Adds n x,y,z triplets to a contour plot. If *replot* is non-zero, the new x,y,z points plus any pre-existing x,y,z points are triangulated. If *replot* = 0, the new x,y,z points are added to the plot's data but are not triangulated and the plot is not redrawn. This is useful in environments (Microsoft Excel in particular) which limit the size of a command string, such that multiple XYZ (or XYZEx) commands are needed.

**Important:** The maximum command string length that DPlot will process is 32768 characters, including the command itself and all delimiters and spaces. For this reason you should limit the number of points sent with XYZEx to a maximum of about 1000 from a VB or C program, for example. If your application requires sending more than 1000 points, use multiple XYZ commands in

#### XJR XViewer

separate XTYP_EXECUTE (LinkExecute in VB jargon) transactions. If you are manipulating DPlot from Microsoft Excel, the maximum command string length is 256 characters.

If **replot** is non-zero and an error in the triangulation occurs (for example if all points are collinear) an error message IS generated. This behavior is different than the XYZ command. In this case DPlot assumes that the calling program knows whether the sent data can be used to construct a valid plot or not, since the caller instructed DPlot to triangulate the data with the **replot** parameter.

#### 5.2.276 XYZRegen command

#### [XYZRegen()]

Triangulates a plot consisting of random 3D points and draws the plot. This command is generally used in conjunction with a series of XYZEx commands in which the *replot* parameter is 0.

#### 5.2.277 Y2AxisLabel command

#### [Y2AxisLabel("string")]

Sets the Y axis label for a second Y axis label to *string*. The label is not displayed unless one of the <u>Multiple Y Axes</u> options is in effect.

#### See also

Title/Axes menu command

#### 5.2.278 Y2TopDown command

#### [Y2TopDown(state)]

For a non-zero *state*, rotates the second Y axis label 180 degrees such that the base of the text faces the plot. The label is not displayed unless one of the <u>Multiple Y Axes</u> options is in effect. This setting is ignored if the Y labels are drawn horizontally (controlled by <u>YAxisHorizontal</u>).

#### 5.2.279 YAxisHorizontal command

#### [YAxisHorizontal(state)]

If *state* is non-zero, the label for the Y axis will be drawn horizontally on all 2D plots. This setting has no effect on 3D projections.

#### 5.2.280 YAxisLabel command

#### [YAxisLabel("string")]

Sets the label for the Y axis to *string*. *string* is limited to 200 characters. For formatting codes (**bold** , *italic*, subscripts, color, etc.) see the description of the Title/Axes menu command.

See also

Title/Axes menu command

#### 5.2.281 YCutoff command

[YCutoff(state,lower,upper)]

#### ✓JR ×Viewer

✓JR ×Viewer

## ✓JR ×Viewer

✓JR XViewer

✓JR XViewer

If *state* is non-zero, sets the *lower* and *upper* amplitude bounds outside of which data points on an XY plot will be ignored. This is equivalent to the <u>Amplitude Limits</u> command on the Options menu.

#### 5.2.282 ZAxisLabel command

#### [ZAxisLabel("string")]

#### ✓JR XViewer

Sets the Z axis label used on 3D contour plots to *string*. Limited to 200 characters. For formatting codes (**bold**, *italic*, subscripts, color, etc.) see the description of the Title/Axes menu command.

See also

Title/Axes menu command

## 5.3 Plugins for File Import, Export and Data Manipulation

Plugins are separate modules (DLL's) that add functionality to DPlot. DPlot supports three types of plugin: file import for file formats not supported by DPlot, file export for both data files and image formats, and "generate" plugins for manipulating existing data in some way. DPlot Jr users please note: DPlot Jr supports file export plugins **only**. It does not support file import or generate plugins.

Sample plugins (file import and generate) are available on the Download page at the main <u>DPlot web</u> <u>site</u>. C source code is provided along with the plugin DLL's. The comments within the source code explain in some detail what is needed; the pointers below are included for emphasis. Constants shown below, e.g. "DATA_3D", are defined in the file dplot.h distributed with the example plugins.

Plugin modules should be placed below the folder where dplot.exe is located (default=c:\Program Files\DPlot) in the \plugins\fileimport, \plugins\fileexport, or \plugins\generate folders. Generate plugins may also be placed in subfolders below \plugins\generate. In that case the subfolder name will be used as a popup menu item on the Generate menu.

#### File Import functions

plugin_init is called by DPlot at startup.

```
___decl spec(dl l export) int pl ugi n_i nit ( char * Desc,
char * Def Ext,
int * Dat aType)
```

Desc A string description of the plugin that will appear in the "Select File Type" dialog. Def Ext A default file extension, including the '.' (return '\0' for none)

Dat aType A data type descriptor. This **must** be DATA_XYXY (XY plot), DATA_3D (Z values on a rectangular grid) or DATA_3DR (random 3d points). If combined with PLUGIN_DOUBLES, then the *Data1 and *Data2 pointers passed to **plugin_readfile** will be for double- rather than single-precision arrays.

Return value from plugin_init should be PLUGIN_INIT_OK plus (optionally) PLUGIN_INIT_MULTIFILE if the user should be allowed to select multiple files to open prior to reading the files (one at a time) with plugin_readfile. Note that DPlot will ignore PLUGIN_INIT_MULTIFILE for 3D data.

**plugin_version** returns the version number of the DPLOT structure expected in subsequent calls to plugin_readfile. This function is **required** for structure versions 3 and later, and optional for earlier versions. See notes below.

```
<u>____decl spec(dl l export) int pl ugi n_version()</u>
```

plugin_readfile reads a file.

__declspec(dllexport) int plugin_readfile( HWND hwndOwner,

char	Filename,
DPLOT	*DPlot,
f I oat	* Dat a1.
f I oat	* Dat a2)

- hwndOwner Handle of the DPlot main window. This window should be disabled if plugin_readfile opens a DialogBox or MessageBox, or performs any other action that allows user input. The window should be re-enabled before exiting.
- Fill ename Full path specification of the file to read.
- DPI ot Structure containing key plot information (labels, array sizes, etc.) This structure is defined in dplot.h (distributed with the example plugins) and also described <u>below</u>. Dat a1 DataType-dependent:
  - DATA_XYXY Pointer to X. You **must** check to insure that the input X and Y arrays are large enough to accomodate the new data. If they are not, the plugin should fail with a return code of PLUGIN_BADARRAYSIZE and return the required array sizes in the DPlot structure. Depending on the context, at that point DPlot will either abort the operation or resize existing arrays and call plugin_readfile again.
    - DATA_3DR Pointer to X,Y,Z triplets. E.g. Data1[0]=x1, Data1[1]=y1, Data1[2] = z1, Data1[3] = x2, etc.
    - DATA_3D Pointer to Z values. Values are arranged such that X varies fastest. On entry, the total allowable number of Z values is (DPlot->MaxCurves+1)* (DPlot->MaxPoints+1). If this is insufficient then the plugin should fail with a return code of PLUGIN_BADARRAYSIZE. Whether successful or not, the grid dimensions should be returned in MaxCurves (number of grid cell columns) and MaxPoints (number of grid cell rows).

Dat a2 DataType-dependent:

DATA_XYXY Pointer to Y. See notes above concerning array sizes. DATA_3DR Unused. DATA_3D Pointer to grid extents, in the order Xlow, Ylow, Xhigh, Yhigh.

Since plugin_readfile may be called for either an empty document or to append data to a document, it must take note of the NumCurves entry in the DPlot structure for DATA_XYXY. The X and Y arrays will already point to the first available curve, but entries in the DPlot structure are for the entire document.

**plugin_readfileex** is identical to plugin_readfile with the addition of an "Options" string in the argument list. A file import plugin should include either **plugin_readfile** or **plugin_readfileex**, but not both. **plugin_readfileex** requires DPlot version 1.9.8.4 or later.

__decl spec(dllexport) int plugin_readfileex( HWD hwndOwner,

char Filename, DPLOT * DPlot, float * Data1, float * Data2, LPSTR Qptions)

Qpt i ons Plugin-defined options, generally a list of comma-separated values. The description of the Options should be included with the plugin distribution. Although how a plugin uses these options is entirely up to the plugin author, this entry is generally used to programmatically accept options rather than displaying a dialog box. For example the "Bitmap To 3D" plugin will use Options if any options are included and, for an incomplete options list, use default options from the last time the plugin was used. Conversely, if no options are specified then the plugin will display its normal input dialog.

For file import plugins distributed with DPlot, these options (when present) are described in the Help topic for the <u>FileOpen</u> command.

**plugin_typecheck** is optional, and informs DPlot whether a file is the correct format for plugin_readfile to interpret it.

__decl spec(dllexport) int plugin_typecheck( char *Filename )

Return value should be:

- -1 Don't know or don't care.
- 0 No, file is not the correct format.
- 1 Yes, file is the correct format.

This info is used for drag-and-drop operations, command line operations, opening a file with Explorer's "Open with..." right-click menu or double-clicking the filename if an association has been made with DPlot. If a file is **not** a DPlot file then DPlot checks with plugin_typecheck for all fileimport plugins for a match.

In all of the above cases (as opposed to opening a file with File>Open), if plugin_typecheck returns 1 then DPlot will, before calling **plugin_readfile**, write an entry to the registry indicating that default values should be used (at the discretion of the plugin). The key will be located at HKEY_CURRENT_USER\Software\DPlot\Plugins\FileImport\<plugin name> and have the value name 'UseDefaults'.

If the plugin checks for and responds to that setting, for example by using default options or the options used in the last call to plugin_readfile, then the plugin should also reset that entry before exiting:

```
if(RegCreateKeyEx(HKEY_CURRENT_USER,
          "Software\\DPlot\\Plugins\\FileImport\\<plugin name>",
          0,
          NULL,
          REG_OPTION_NON_VOLATILE,
          KEY_ALL_ACCESS,
          NULL,
          &hKey,
          &dwDisposition) == ERROR_SUCCESS)
{
          RegSetValueEx(hKey,"UseDefaults",0,REG_SZ,"0",1);
          RegCloseKey(hKey);
}
```

}

#### File Export functions

plugin_init is called by DPlot at startup.

```
__decl spec(dllexport) int plugin_init( char *Desc,
char *Def Ext,
int *DataType)
```

Desc A string description of the plugin that will appear in the "File Formats" dialog.

Def Ext A default file extension, including the '.' (return '\0' for none)

Dat aType A data type descriptor. Valid values are DATA_XYXY (XY plot), DATA_3D (Z values on a rectangular grid), DATA_3DR (random 3d points), or DATA_IMAGE (bitmap image). On export, DPlot calls **plugin_writeimage** for DATA_IMAGE plugins; for all others it calls **plugin_writefile**. If combined with PLUGIN_DOUBLES, then the *Data1 and *Data2 pointers passed to **plugin_writefile** will be for double- rather than single-precision arrays.

The return value of plugin_init should always be PLUGIN_INIT_OK.

**plugin_version** returns the version number of the DPLOT structure expected in subsequent calls to plugin_writefile. This function is **required** for structure versions 3 and later, and optional for earlier

versions. See notes below.

__decl spec( dl l export ) int pl ugi n_versi on( )

#### plugin_writefile writes a data file.

__decl spec(dllexport) int plugin_writefile( HMND hwndOwner,

char	Filename,
DPLO	「*DPlot,」
f I oat	* Dat a1,
float	* Dat a2)

- hwndOwner Handle of the DPlot main window. This window should be disabled if plugin_writefile opens a DialogBox or MessageBox, or performs any other action that allows user input. The window should be re-enabled before exiting.
- Fil ename Full path specification of the file to write.
- DPI ot Structure containing key plot information (labels, array sizes, etc.) This structure is defined in dplot.h (distributed with the example plugins) and also described <u>below</u>. Dat a1 DataType-dependent:
- DataType-dependent. DATA_XYXY Pointer to X. DATA_3DR Pointer to X,Y,Z triplets. E.g. Data1[0]=x1, Data1[1]=y1, Data1[2] = z1, Data1[3] = x2, etc. DATA_3D Pointer to Z values. Values are arranged such that X varies fastest. DataType-dependent:
  - DATA_XYXY Pointer to Y.
  - DATA_3DR Unused.

DATA_3D Pointer to grid extents, in the order Xlow, Ylow, Xhigh, Yhigh.

**plugin_writeimage** writes a bitmap image of the currently active document window. **Please note:** This function is still supported but is obsolete and has been replaced with **plugin_writeimage_ex** (see below).

decl spec( dl I export )	i nt	plugin_writeimage(	HWD hwndOwner, char Filename, HBITMAP hbmp, HPALETTE hpal)

- hwndOwner Handle of the DPlot main window. This window should be disabled if plugin_writefile opens a DialogBox or MessageBox, or performs any other action that allows user input. The window should be re-enabled before exiting.
- Fil ename Full path specification of the file to write.
- hbmp Handle of a device-dependent bitmap. Bitmap dimensions, color depth, and other settings are specified by the user (Image Properties dialog) or by a SetPluginImageDims macro/DDE command.
- hpal Handle to the palette used on the active plot. This palette is valid even though it may not actually be used in the bitmap. For example, 24- and 32-bit color bitmaps do not use a palette, but the majority of the colors within the bitmap are all found within the palette.

___declspec(dllexport) int plugin_writeimage_ex(HWND hwndOwne

HVWD	nwr	nacwne	er i
char	Fi I	enam	Э,
HBI TI	WAP	hbmp.	
HPAL	ETTE	E hp'al	
int '	* opt	i ons)	ŕ
		/	

Your plugin should include **either** plugin_writeimage or plugin_writeimage_ex, but not both. All parameters are as described above with the addition of:

- opt i ons Array of 4 integers containing additional image options:
- opt i ons[0] Indicates desired bit depth, 8 or 24 bits per pixel.
- opt i ons[1] Compression option. Currently this setting is used only to indicate whether RLE encoding should be used for 8-bit BMP files.

- opt i ons[2] Resolution, in dpi. Some programs may use this setting to determine the physical size that the image should be scaled to, particularly when printing.
- opt i ons[3] Not defined and cannot be set interactively within DPlot, but can be set with a <u>SetPluginImageDims</u> macro/DDE command. This field might be useful for a custom image export plugin.

#### **Generate** functions

plugin_init is called by DPlot at startup.

decl spec	c(dllexport) int plugin_init( char *Desc, char *Uhused, int *Action )
Desc Action	<ul> <li>String description of the plugin that will appear in the Generate menu.</li> <li>An "action" code that describes how plugin_run operates on the input data. Value should be one of (defined in dplot.h):</li> <li>PLUGIN_REPLACEDATA <ul> <li>Plugin operates on and replaces existing data in the same document.</li> </ul> </li> <li>PLUGIN_NEWDATA <ul> <li>Plugin operates on existing data and returns one or more data sets (curves) for the same document.</li> </ul> </li> <li>PLUGIN_NEWDOCUMENT <ul> <li>Plugin operates on existing data (or not) and returns an entirely new document. In this case the input provided by DPlot is a copy of an existing document – the plugin does not have to concern itself with opening a new window.</li> </ul> </li> <li>If combined with PLUGIN_DOUBLES, then the *X and *Y pointers passed to plugin_run will be for double- rather than single-precision arrays.</li> </ul>

Return value for plugin_init should be PLUGIN_INIT_OK.

**plugin_version** returns the version number of the DPLOT structure expected in subsequent calls to plugin_run. This function is **required** for structure versions 3 and later, and optional for earlier versions. See notes below.

<u>___decl spec(dl l export) int pl ugi n_version()</u>

plugin_run modifies existing data or generates new data.

decl spec	(dllexport) int	pl ugi n_r un(	HVND hwnde DPLOT * DPI f I oat * X,	Owner, ot, float	* Y)
hwndOwner	Handle of the DPk	ot main window.	This window	v should	be disable

- hwndOwner Handle of the DPlot main window. This window should be disabled if plugin_run opens a DialogBox or MessageBox, or performs any other action that allows user input. The window should be re-enabled before exiting.
- DPI ot Structure containing key plot information (labels, array sizes, etc.) This structure is defined in dplot.h (distributed with the example plugins and also described <u>below</u>).
- X, Y X and Y arrays. For actions that create new data values, you **must** check to insure that the input X and Y arrays are large enough to accomodate the new data. If they are not, the plugin should fail with a return code of PLUGIN_BADARRAYSIZE and return the required array sizes in the DPlot structure. Depending on the context, at that point DPlot will either abort the operation or resize existing arrays and call plugin_run again.

**plugin_run_cmds** is identical to plugin_run with the addition of an "Options" string in the argument list. A generate plugin should include either **plugin_run** or **plugin_run_cmds**, but not both. **plugin_run** requires DPlot version 2.3.2.1 or later.

__decl spec(dllexport) int plugin_run_cmds( H/WD hwndOwner, DPLOT * DPlot, float * X, float * Y, LPSTR Options)

Qpt i ons Plugin-defined options, generally a list of comma-separated values. The description of the Options should be included with the plugin distribution. Although how a plugin uses these options is entirely up to the plugin author, this entry is generally used to programmatically accept options rather than displaying a dialog box. If no options are specified then the plugin will display its normal input dialog.

For generate plugins distributed with DPlot, these options (when present) are described in the Help topic for <u>Plugin modules</u>.

#### **DPLOT Structure**

This defined type is used in calls to <u>plugin_readfile</u>, <u>plugin_readfileex</u>, <u>plugin_writefile</u>, and <u>plugin_run</u>. The DPLOT structure is defined in dplot.h, included with the example plugins and repeated here:

#define DPLOT VERSION 4 // Version 4 of the DPLOT structure adds a 4th title line and allows up to 200 characters for each title 11 line 11 and axis label. It requires version 2.1 or later of 11 DPI ot. Older structure versions are defined in dpl ot . h #define MAXC 100 // Maximum number of curves in an XY plot. Do not change. // These are the only data types supported by the plugin interface: #define DATA_XYXY 0 // XY data // Z values on a rectangular grid // Random X, Y, Z values #define DATA_3D 3 ã #define DATA_3DR typedef struct tagDPLOT DWORD Version: // version number of this structure DWORD hwnd; // handle of client application window (Use DWORD rather than HWND) 11 DWORD Dat a For mat; // DATA_XYXY, DATA_3D, or DATA_3DR DWORD MaxOur ves; // maximum number of curves (must be <= 100) // = NK for DataFormat = DATA_3D // ignore for DataFormat =  $\overline{D}ATA_3DR$ // maximum number of points/curve DWORD MaxPoints; // = NY for DataFormat = DATA_3D // = 3 * number of points for DATA_3DR
// actual number of curves, always 1 for
// DATA_3D or DATA_3DR DWORD NumOur ves; // scaling code (Linear, Log, etc.) // left coord of legend, expressed as a ratio DWORD Scal e: float LegendX; 11 of plot size (0->1)float LegendY // top coord of legend actual number of points in each curve; DWORD NP MAXCI; 11 cannot exceed MaxPoints. For DATA_3DR files, return number of nodes 11 11 11 in NP[0] DWCRD LineType[MAXC]; DWCRD Symbol Type[MAXC] 11 line types (see codes in dplot.h) symbol types (see codes in dplot.h) Address of 32K character string that may be used to pass command strings back to 11 LPSTR I pszCommands; // 11 11 DPI ot. 11 NOTE: Requires DPI ot version 2.0.7 or later. Earlier versions of DPlot will 11 pass | pszCommands == NULL. Attempting to 11

	//	use lpszCommands in this case will result
	//	in a crash.
char	Legend[ MAXC+1] [ 80] ;	
	//	Legend[0] is the caption for the legend.
	//	Legendini is the legend for the n'th curve.
char	Label [ MAXC] [ 40] ; //	Strings displayed beside the last data
	//	point in a curve.
char	Title[4][200]; //	Four title lines.
char	XAxis 12001: 1/	X Axis I abel .
char	YAxi s 200 ( //	YAxis label.
DPLOT;		

<u>Plugin readfile</u>, <u>plugin readfileex</u>, and <u>plugin run</u> functions can pass DPlot command strings back to DPlot with the I pszCommands member. These commands are identical to those used in DPlot macros and/or in calls to the <u>DPLOTLIB.DLL</u> function DPlot_Command. For more information on DPlot command strings see <u>Sending data to DPlot from another application</u>.

#### **NOTES**

}

Plugins should be placed in the appropriate folder ("fileexport", "fileimport" or "generate") below <dplot folder>\plugins.

Function <u>plugin version</u>: Newer versions of the DPLOT structure reflect a few of DPlot's newer capabilities, namely the ability to plot up to 100 curves (instead of 20) and use more characters in the legend, title lines, and axis labels. The change to the structure requires an additional function (plugin_version) to tell DPlot to send the address of the newer structure. Old plugins written for version 2 of the DPLOT structure will continue to work with newer versions of DPlot. New plugins using version 3 or 4 of the structure will not work with older version 1.8 or later of DPlot if you intend to use the version 3 structure, or version 2.1 or later of DPlot for the version 4 structure. You will also need to ensure that your plugin gracefully exits if the wrong structure version is passed (which will happen if the user has a version of DPlot older than 1.8). An example of this is included in the example plugins.

**If you write** a plugin you believe would be useful to others, <u>let us know</u> and it will be added to the DPlot download page.

Not a programmer? <u>Registered users</u> may request a new plugin to support additional file formats or data manipulation operations.



## 6 Reference

Keyboard Shortcuts Right-Click Mouse Operations Functions Error Messages Known Problems Acknowledgements

## 6.1 Keyboard Shortcuts

DPlot has a set of keyboard shortcuts that you can use to carry out commands and select dialog box options directly from the keyboard - by using these shortcut keys, you bypass the menus. With other shortcut keys, you can move, select, and edit large documents more efficiently.

Keys that must be pressed sequentially are shown separated by a comma; keys that must be pressed at the same time are shown separated by a plus sign.

Help keys To	Press
Switch to Help window	F1
Switch to context-sensitive Help	SHIFT+F1
Editing To	Press
Carry out an action	ENTER
Cancel an action	ESC
Clear the selection	DELETE
Copy the active document to the Clipboard	CTRL+C
Paste data from the Clipboard	CTRL+V
Working with files	Brocc
Create a new degument	
Create a new document	
command on the File menu	CIRL+5
Open an existing file	CTRL+O
Print the active document	CTRL+P
Close the active document	CTRL+F4
Quit DPlot	ALT+F4

# Working with windows To

Select the menu bar	ALT or F10
Select the DPlot Control menu	ALT+SPACEBAR
Select the document Control menu	ALT+HYPHEN
Close the document window	CTRL+F4
Switch to the next open document window	CTRL+F6
Switch to the previous open document window	CTRL+SHIFT+F6
Tile document windows vertically	SHIFT+F4
Tile document windows horizontally	SHIFT+F6
Arrange all open documents as overlapped windows with the active document on top.	SHIFT+F5
Zoom all (show entire plot extents after zooming in)	CTRL+1
Zoom in on the current mouse position by a factor of 2 for XY plots and 2D views of 3D data.	CTRL+2 or CTRL++ (Plus sign on the numeric keypad)
Zoom in on the current mouse position by a factor of 3, 4, or 5 for XY plots and 2D views of 3D data.	CTRL+3, CTRL+4, CTRL+5
Zoom out	CTRL+- (CTRL key and the minus sign on the numeric keypad)
Within dialog boxes with text formatting c	odes

Press

Within dialog boxes with text formatting codes	
То	Press
Subscript	CTRL+₊
Superscript	CTRL++
Insert Symbol	CTRL+S
Bold	CTRL+B
Italic	CTRL+I
Underline	CTRL+U
Double size	CTRL+2
Color	CTRL+K

# 3D Views of 3D/4D data:

То	Press
Decrease azimuth by 1 degree	CTRL+≁
Increase azimuth by 1 degree	CTRL+ →
Decrease elevation by 1 degree	CTRL+∔
Increase elevation by 1 degree	CTRL++

Decrease azimuth by 5 degrees	CTRL+SHIFT+ ←
Increase azimuth by 5 degrees	CTRL+SHIFT+→
Decrease elevation by 5 degrees	CTRL+SHIFT+∔
Increase elevation by 5 degrees	CTRL+SHIFT++

**Please note:** These shortcuts work best with simple plots: no more than a few thousand data points, <u>antialiasing</u> and <u>lighting</u> turned off. For the fastest rendering, use a wireframe rendering and turn off antialiasing using the right-click menu:

	Contour Options
	Extents/Intervals/Size
	Generate Mesh
	Triangular Mesh
	Surface Plot
	Waterfall Plot
✓	Wireframe
	Bar Chart
	Turn Antialiasing.Off
	3

Once you have the view angles you want, you can then quickly switch back to a normal surface plot and add antialiasing if you like with this same right-click menu.

## 6.2 Right-Click and Mouse Wheel Operations

Many simple operations within DPlot can be performed with a right-click of the mouse or a turn or click of the mouse wheel. This topic illustrates all of the areas of a plot in which various right-click and mouse wheel operations may be performed.

## **Right-Click**

For many of the options discussed in this topic, we'll take a look at the example DPlot file EX10.GRF.
### **Scaling**

Freefall in a Vacuum 450 137.16 =V_ot - gt²/2 Initial Velocity, Vo ٧ =V_/g **)-** 150 fps t_{max} 400 121.92  $y_{max} = V_0^2/2g$ 349.6612 ft 100 fps 50 fps 350 106.68 Linear X - Linear Y 300 Linear X - Logarithmic Y 91.44 Logarithmic X - Linear Y Logarithmic X - Logarithmic Y 250 76.2 Meters Tripartite Grid Feet Linear X - Probability Y 200 60.96 Grain Size Distribution 155.40 Polar Coordinates Bar Chart 150 45.72 Logarithmic X - Probability Y Probability X - Linear Y 100 30.48 Probability X - Logarithmic Y Probability X - Probability Y 38.85125 ft 50 Triangle Plot 15.24 N1.85 Hydraulic curve Mercator Projection 0 2 4 5 6 7 9 10 3 8 Time, seconds

Right-click anywhere within an unoccupied area of the boundaries of a graph to display axis scaling options.

#### Symbol and Line Styles, Color, Copying data, Hiding a curve, Adding a Label



Right-click on a curve to change the line style, symbol style, line width, symbol size or color used to draw the curve, copy tab-separated columns of data to the Clipboard, or hide a curve. (Tip: you can also hide a curve w/o the right-click menu by pointing to a curve and clicking the middle mouse button, or hide all curves **other** than the one clicked by pressing CTRL + the middle mouse button. Clicking the middle mouse button while the mouse does **not** point to a curve will make all curves visible.)



Units for the line width and symbol size menues are determined by the *Default dimensions* setting on the <u>General Options</u> dialog.

If you hide a curve using the right-click menu, you can later make it visible with the **Hide Curves** or **Unhide All Curves** commands on the View menu.



Select "Add \$X, \$Y Label" to add a text label tied to a specific data point. The \$X will be replaced with the corresponding X value; \$Y will be replaced with the corresponding \$Y value. Both X and Y will be rendered using the number format for the respective axis. The label will initially have a frame and be opaque, middle-right justified at the point or middle-left justified for points near the right edge of the document window. If a point is already labeled, this option is not shown.



You can drag the label with your mouse to a new location and change any of the label's attributes

with the right-click menu (see below) or double-click on the label to bring up the Point Labels dialog box.

#### Title lines and axis labels



Right-click on any title line or the axis labels to edit the text or to change the font of that character string.

#### Legend



Right-click anywhere within the legend to change its attributes, hide it, or change how the legend box is aligned with its specified location. If the legend is hidden then this right-click menu is of course no longer accessible. In that case you can edit legend attributes with the <u>Legend/Labels</u> command on the Text menu.

#### Legend entries



Right-click on specific legend entries to hide the corresponding curve, hide all curves **except** for the corresponding curve, and/or to change the precedence of a curve. Curves hidden with the right-click menu can later be made visible with the **Hide Curves** or **Unhide All Curves** commands on the View menu.

#### <u>Notes</u>



Right-click on a note to edit the text, hide the note, change its attributes, or change how the note is aligned with its specified location. (If a note is hidden then the right-click menu is of course no longer accessible and note attributes can only be changed with the <u>Add/Edit Note</u> command on the Text menu.)

#### **Point labels**



Right-click on a point label to edit the text, delete the label, or change its attributes. "Copy" copies the text of the label to the Clipboard, replacing text shortcuts (e.g. \$X, \$Y) with their numeric substitutions.



Right-click on any of the numbers along an axis to change the format used for those numbers, set the extents of that axis, or change the tick mark interval.

#### Axis style



Right-click on the coordinate axes or the box around the plot to change the style of the axes.

#### Edit menu

		Freerair in a
	Undo Line width	Ctrl+Z
	Redo	Ctrl+Y
	Сору	+
	Paste	Ctrl+V
	Paste Special	
	Insert Link	N
	Move/Copy Curves	
	Combine Curves	-
2	Operate on X	
	Operate on Y	-
eet	Decimate Points	•
۲	Delete Points in Box	
	Edit Data	

Right-click outside the extents of a graph to display the Edit menu.

x10.arf



You can quickly toggle the <u>Print Caption</u> feature of the Text menu on/off by right-clicking in the lower right corner of a plot.

#### **3D Options**



For all 3D data you can display most of the Options menu options by right-clicking anywhere on the plot. While displaying 3D views this menu will include options for various applicable plot types (Waterfall Plot and Bar Chart are not shown for random 3D points; those options are only valid for Z values on a rectangular grid. To produce a table of Z values from random 3D points see the <u>Generate Mesh</u> command on the Options menu.) The latter options are particularly useful for spinning a plot around using the <u>Ctrl+arrow keys shortcuts</u>: Surface plots generally take much longer to draw than any other plot type because DPlot calculates the color of every individual pixel. So you might want to switch to a wireframe plot before using those shortcuts, then switch back to a surface plot using this same right-click option once you have the view angles you want.



For 2D views of surface plots with contour lines, you can right-click on any of the contour labels to change the font, color, transparency, and frequency of those labels. Each of these options is of course also available on the <u>Contour Options</u> dialog.

# **Mouse Wheel**

#### <u>Zoom</u>

You can zoom in by a factor of 2 by rotating the mouse wheel forward, and undo previous zooms by rotating the mouse wheel backward. When zooming in, new extents are based on the current mouse position (the new center of the graph will be roughly located at the mouse coordinates). If the mouse is not within the plot, the new extents are based on the center of the plot.

You can restrict zooming to the X direction by simultaneously pressing the CTRL key, or to the Y direction by pressing the SHIFT key. If you find that you often unintentionally zoom in or out by accidentally rotating your mouse wheel, you can disable this feature with the **Disable zooming** with mouse wheel checkbox under the <u>General Options</u> menu command.

#### Hide/Unhide Curves

For XY plots, you can:

- hide a curve by clicking the middle mouse button while the mouse points to a curve.
- hide all curves except the one pointed to by pressing CTRL + the middle mouse button.
- make all curves visible by clicking the middle mouse button while the mouse does not point to a curve.

If you find that you often unintentionally hide/unhide curves by accidentally pressing the middle mouse button, you can disable this feature with the **Disable hide/unhide with middle mouse button** checkbox under the <u>General Options</u> menu command.

# 6.3 Functions

This table provides a brief description of the functions available in the  $\underline{Y=f(X)}$  command and similar commands. For a more detailed description of each function, click on the function name. Function names are **not** case specific.

For functions that take multiple arguments, the arguments must be separated by a comma, or, if the <u>decimal symbol</u> is set to a comma, by a semicolon.

Please note: All dates taken as input arguments to these functions are Excel 1900 date system serial numbers, the internal date format used throughout DPlot.

ABS	Absolute value
ACOS	Arccosine (inverse cosine)
ACOSH	Inverse hyperbolic cosine
AND	Bitwise AND
ASIN	Arcsine (inverse sine)
ASINH	Inverse hyperbolic sine
ATAN	Arctangent (inverse tangent)
ΔΤΔΝ2	Arctangent
	Inverse hyperbolic tangent
RESSELL	Bessel function of the first kind
BESSELV	Bessel function of the second kind
	Cumulative distribution function
	Rounds a number up
	Cosine
COSH	Hyperbolic cosine
DATE	Excel 1900 date system serial number that represents the input
	year, month, and day
<u>DAY</u>	Day of the month from a date (1-31)
ERF	Error function
ERFC	Complementary error function
EXP	Exponent (inverse of log)
FACT	Factorial
FLOOR	Rounds a number down
FPOWER	Raises a number to a fractional power
GCD	Greatest common divisor
LCM	Least common multiple
	Natural (base-e) logarithm
LOG10	Rase-10 logarithm
MAX	Maximum of two values
MIN	Minimum of two values
	Pomainder after division
	from a data (1- lanuary)
	Pounds a number to a specified multiple
	Normal distribution
	Normal distribution
NURIVIDISTCUE	Cumulative distribution function for the specified mean and
	standard deviation.
NORMINV	Inverse of the cumulative distribution function
NORMRAND	Normally-distributed random number for a distribution with a
	specified mean and standard deviation.
NOT	Bitwise NOT
<u>OR</u>	Bitwise OR
<u>PMT</u>	Calculates the payment for a loan based on constant payments
	and a constant interest rate.
RANDBETWEEN	Inumber between specified limits
ROUND	Rounds a number to a specified number of digits
RND	Random whole number
SIGN	of a number
SIN	Sine
SINH	Hyperbolic sine
SORT	Positive square root
SLIM	Add result of expression to previous result (Taylor series for
	avampla)
	example)

TAN	Tangent
TANH	Hyperbolic tangent
WEEKDAY	Day of the week from a date (1=Sunday, 7=Saturday)
WEEKNUM	Number of weeks since start of year for a given date
XOR	Bitwise XOR
YEAR	Year from a date

# 6.3.1 ABS function

Returns the absolute value of a number. The absolute value of a number is the number without its sign.



See also: Functions

# 6.3.2 ACOS function

Returns the arccosine, or inverse <u>cosine</u>, of the given number. The result will be in radians or degrees, depending on the Angles options. With "degrees" selected, acos(-1) returns 180.



COS function Functions

#### FUNCTIONS

# 6.3.3 ACOSH function

Returns the inverse of the hyperbolic cosine (cosh).

$$a \cosh(x) = \log(x + \sqrt{x+1}\sqrt{x-1})$$

The input argument must be  $\geq$  1.



See also: ASINH function ATANH function COSH function Functions

## 6.3.4 AND function

Returns the result of the logical AND operation on the corresponding bits of two arguments. Both arguments are first truncated to integers; the result will always be an integer value. In C syntax, the result of AND(a,b) is equivalent to (floor(a) & floor(b)).

Example:



XOR function

# 6.3.5 ASIN function

Returns the arcsine, or inverse <u>sine</u>, of the given number. The result will be in radians or degrees, depending on the Angles options. With "degrees" selected, asin(1) returns 90.



ACOS function SIN function Functions

# 6.3.6 ASINH function

Returns the inverse of the hyperbolic sine (sinh).





# 6.3.7 ATAN function

Returns the arctangent, or inverse <u>tangent</u>, of the given number. The result will be in radians or degrees, depending on the Angles options. ATAN returns a value in the range –PI/2 to PI/2 radians (or -90 to 90 degrees). With "degrees" selected, atan(1) returns 45.



# 6.3.8 ATAN2 function

**atan2(y,x)** returns the arctangent, or inverse <u>tangent</u>, of y/x. The result will be in radians or degrees, depending on the Angles options. ATAN2 returns a value in the range –PI to PI radians (or -180 to 180 degrees), using the signs of both parameters to determine the quadrant of the return value.

See also: ATAN function Functions

## 6.3.9 ATANH function

Returns the inverse of the hyperbolic tangent (tanh).

$$a \tanh(x) = \frac{1}{2}\log\left(\frac{1+x}{1-x}\right)$$

for -1 < x < 1



### 6.3.10 BESSELJ function

**BESSELJ(x,order)** returns the Bessel function of the first kind at **x** for the specified **order**. The Bessel functions of the first kind are defined as the solutions to the Bessel differential equation

$$x^{2} \frac{d^{2} y}{dx^{2}} + x \frac{dy}{dx} + (x^{2} - n^{2})y = 0$$

which are nonsingular at the origin. They are sometimes also called cylinder functions or cylindrical harmonics.

The n-th order Bessel function of x is:

$$J_{n}(x) = \sum_{k=0}^{\infty} \frac{(-1)^{k}}{k! \Gamma(n+k+1)} \left(\frac{x}{2}\right)^{n+2k}$$

where:

$$\Gamma(n+k+1) = \int_{0}^{\infty} e^{-k} x^{n+k} dx$$

is the Gamma function.

The plot below shows the Bessel function for n=0,1,2,3,4,5.



**Order** must be greater than or equal to 0. If **order** is not an integer, it is truncated. **x** must be greater than or equal to 0.

For more information on the Bessel function see the reference at MathWorld



# 6.3.11 BESSELY function

**BESSELY(x,order)** returns the Bessel function of the second kind at **x** for the specified **order**. The Bessel functions of the second kind are defined as the solutions to the Bessel differential equation

$$x^{2} \frac{d^{2} y}{dx^{2}} + x \frac{dy}{dx} + \left(x^{2} - n^{2}\right)y = 0$$

which are singular at the origin.

The n-th order Bessel function of x is:

$$Y_{\mathbf{x}}(x) = \lim_{\mathbf{v} \to \mathbf{x}} \frac{J_{\mathbf{v}}(x) \cos(\mathbf{v}\pi) - J_{\mathbf{v}}(x)}{\sin(\mathbf{v}\pi)}.$$

where J denotes the Bessel function of the first kind.

The plot below shows the Bessel function for n=0,1,2,3,4,5.



**Order** must be greater than or equal to 0. If **order** is not an integer, it is truncated. **x** must be greater than or equal to 0.

For more information on the Bessel function see the reference at MathWorld



# 6.3.12 CDF function

(Cumulative Distribution Function) Useful when used in conjunction with probability scales. If x is the number of standard deviations a measurement departs from the mean, then the associated probability (0-100) is:

$$50 * \left[ 1 + \operatorname{erf} \left( \frac{\chi}{\sqrt{2}} \right) \right]$$

cdf(x) will plot as a straight line on a probability scale.

cdf((x-<mean>)/sqrt(<stdev>)) will give the cumulative distribution function for a series with mean value=<mean> and standard deviation <stdev>.



**Cumulative Distribution Function** 

ERF function Functions

## 6.3.13 CEIL function

Returns the smallest integer that is greater than or equal to the input number.

CEIL(4.2)=5 CEIL(-4.1)=-4.

See also: FLOOR function

**Functions** 

# 6.3.14 COS function

Returns the cosine of the given angle. The angle may be in radians or degrees, controlled by the Angles options. With "radians" selected, cos(pi) returns -1.0





# 6.3.15 COSH function

Returns the hyperbolic cosine of the number.



See also: ACOSH function SINH function Functions

## 6.3.16 DATE function

Returns the Excel 1900 date system serial number that represents the input year, month, and day. For example,

date(2005,2,18) returns 38401 (the 38,401st day since January 1, 1900).

Months are numbered 1-12 (January through December). Day is 1-based (1 is the first day of a month).

### Syntax

DATE(year,month,day)

See also:

DAY function MONTH function WEEKDAY function WEEKNUM function YEAR function Functions

## 6.3.17 DAY function

Returns the day of a date represented by an Excel 1900 date system serial number. The day is given as an integer between 1 and 31.

### Syntax

DAY(serial_number)

See also: DATE function MONTH function WEEKDAY function WEEKNUM function YEAR function Functions

# 6.3.18 ERF function

Returns an approximation of the error function:

$$erf(x) = \frac{2}{\pi} \int_{0}^{x} e^{-t^{2}} dt$$
$$erf(-x) = -erf(x)$$

For  $abs(x) \le 1$ , DPlot uses the approximation:

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{n!(2n+1)}$$
$$= \frac{2}{\sqrt{\pi}} \left( x - \frac{x^3}{3} + \frac{x^5}{10} - \frac{x^7}{42} + \frac{x^9}{216} + \dots \right)$$

and iterates until any additional term is less than 1.E-10 times the current solution. In general the series converges within several steps for small x and takes up to 15 terms before converging to the solution for x close to 1.0.

For abs(x) > 1 DPlot uses an approximating 9th order polynomial that is accurate to +/- 1.E-7 for all values of x:

```
z = abs(x)
t = 1./(1.+0.5*z)
erf = 1.-t*exp(-z*z -1.26551223 +
t*( 1.00002368 +
t*( 3.7409196E-1 +
t*( 9.678418E-2 +
t*(-1.8628806E-1 +
t*( 2.7886807E-1 +
t*( -1.13520398 +
t*( 1.48851587 +
t*( -8.2215223E-1 +
t* 1.7087277E-1 )))))))))
```

#### See also:

CDF function ERFC function Functions

### 6.3.19 ERFC function

Returns an approximation of the complementary error function. The complementary error function is defined as

ERFC(X)=1 - ERF(X)

See also:

CDF function ERF function Functions

# 6.3.20 EXP function

Returns e raised to the power of the number. The constant e equals 2.71828182845904, the base of the natural logarithm. **exp** is the inverse of **log**, the natural logarithm.

See also: LOG function Functions

## 6.3.21 FACT function

Returns the factorial of a number. The factorial of a number is equal to 1*2*3*...* number. If the number is not an integer, it is truncated. A negative argument will generate an error. Note that if used alone (as in Y=fact(X)), fact will cause a trapped overflow with an argument greater than or equal to 171.

Example:

FACT(4)=24

FACT(20)=2.342902E+18

See also: Functions

# 6.3.22 FLOOR function

Returns the largest integer that is less than or equal to the input number.

FLOOR(56.9)=56 FLOOR(-13.4)=-14.

See also: CEIL function

**Functions** 

# 6.3.23 FPOWER function

Returns the result of the first argument raised to a fractional power. FPOWER(x,a,b) =  $x^{(a/b)}$ .

Both **a** and **b** must be integers (whole numbers). This function is primarily useful for finding fractional powers of negative numbers. (The normal exponentiation operator (^) will not work in this case, as internally x^a is calculated as exp(a*ln(x)), and the logarithm of a negative number is undefined.)

The function returns an invalid argument error if:

b is equal to 0.

x is less than 0, b is even, and a is odd.

FPOWER(-27,1,3)=-3

See also: EXP function LOG function Functions

# 6.3.24 GCD function

Returns the **greatest common divisor** of two integers. The greatest common divisor is the largest integer that divides both numbers without a remainder. If either number is not an integer, it is truncated. The GCD function will always return a positive result, regardless of whether one or both numbers are negative.

GCD(18,84)=6 GCD(7,0)=7 GCD(36,48)=12

See also: LCM Functions

## 6.3.25 LCM function

Returns the **least common multiple** of two integers. The least common multiple is the smallest positive integer that is a multiple of both arguments. If either argument is not an integer, it is truncated.

LCM(a,b) is calculated as  $(a^{*}b)/GCD(a,b)$ .

LCM(5,2)=10

LCM(24,36)=72

See also: GCD Functions

_____

# 6.3.26 LOG function

Returns the natural logarithm of a number. Natural logarithms are based on the constant e (2.71828182845904). **log** is the inverse of the **exp** function.

See also: EXP function Functions

## 6.3.27 LOG10 function

Returns the base-10 logarithm of a number. log10(10)=1, log10(1E5)=log10(10^5)=5.

See also: Functions

# 6.3.28 MAX function

Returns the maximum of two comma-separated values. max(4,7)=7.

See also: <u>MIN function</u> Functions

## 6.3.29 MIN function

Returns the minimum of two comma-separated values. min(4,7)=4.

See also: MAX function

#### **Functions**

#### 6.3.30 MOD function

Returns the remainder after a number is divided by divisor. The result has the same sign as divisor.

#### Syntax

MOD(number, divisor)

Number is the number for which you want to find the remainder.

Divisor is the number by which you want to divide number.

See also: Functions

## 6.3.31 MONTH function

Returns the month of a date represented by an Excel 1900 date system serial number. The month is given as an integer, ranging from 1 (January) to 12 (December).

### Syntax

MONTH(serial_number)

See also:

DATE function DAY function WEEKDAY function WEEKNUM function YEAR function Functions

### 6.3.32 MROUND function

Returns a number rounded to the desired multiple.

#### Syntax

MROUND(number, multiple)

*number* is the value to round.

*multiple* is the multiple to which you want to round number.

MROUND rounds up, away from zero, if the remainder of dividing number by multiple is greater than or equal to half the value of multiple.

 $\mathsf{MROUND}(10, 3) = 9$ 

MROUND(-10, -3) = -9

MROUND(1.3, 0.2) = 1.4

See also: ROUND Functions

## 6.3.33 NORMDIST function

Returns the normal distribution for the specified mean and standard deviation.

#### Syntax

NORMDIST(x,mean,standard_dev)

where

X

is the value for which you want the distribution.

*Mean* is the arithmetic mean of the distribution.

Standard_dev is the standard deviation of the distribution. (must be greater than 0.)

The result of normdist is

$$f(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\left(\frac{(x-\mu)^2}{2\sigma^2}\right)}$$

DPlot's NORMDIST function is equivalent to Excel's NORMDIST with the *cumulative* parameter set to FALSE.

See also: NORMDISTCDF Functions

## 6.3.34 NORMDISTCDF function

Returns the cumulative distribution function for the specified mean and standard deviation.

#### Syntax

```
NORMDISTCDF(x,mean,standard_dev)
```

where X

is the value for which you want the distribution.

*Mean* is the arithmetic mean of the distribution.

Standard_dev is the standard deviation of the distribution (must be greater than 0.)

The result of normdistcdf is the integral of the <u>NORMDIST</u> function from negative infinity to x. It can also be represented by:

$$\frac{1}{2} \left[ 1 + \operatorname{erf} \left( \frac{x - \mu}{\sigma \sqrt{2}} \right) \right]$$

NORMDISTCDF is equivalent to Excel's NORMDIST function with the *cumulative* parameter set to TRUE.

See also: ERF NORMDIST NORMINV

#### **Functions**

#### 6.3.35 NORMINV function

Returns the inverse of the cumulative distribution function for the specified mean and standard deviation.

Syntax

NORMINV(probability,mean,standard_dev)

where

**Probability** is a probability corresponding to the normal distribution (must be  $\geq 0$  and  $\leq 1$ ).

*Mean* is the arithmetic mean of the distribution.

Standard_dev is the standard deviation of the distribution (must be greater than 0.)

The result of norminv is the inverse of the <u>NORMDISTCDF</u> function, i.e. NORMINV(NORMDISTCDF (x,5,2),5,2)=x.

See also: NORMDIST NORMDISTCDF Functions

### 6.3.36 NORMRAND function

Generates a normally-distributed random number from a distribution with the specified mean and standard deviation.

#### Syntax

NORMRAND(mean, standard_dev)

*Mean* is the arithmetic mean of the distribution.

Standard_dev is the standard deviation of the distribution.

#### Example

100,000 calls to NORMRAND(2.3,1.5) produces this plot:



Which yields this histogram (Histogram command on the Generate menu).



#### Mean=2.31002, Standard Deviation=1.50083, Skewness=-0.0133169

# 6.3.37 NOT function

**Functions** 

Returns the result of the bitwise NOT operation on the input argument. The argument is first truncated to an integer; the result will always be an integer value. In C syntax, the result of NOT(a) is equivalent to ~ (floor(a)).

Example:



# 6.3.38 OR function

Returns the result of the logical inclusive OR operation on the corresponding bits of two arguments. Both arguments are first truncated to integers; the result will always be an integer value. In C syntax, the result of OR(a,b) is equivalent to (floor(a) | floor(b)).

Example:



AND function NOT function XOR function

# 6.3.39 PMT function

Calculates the payment for a loan based on constant payments and a constant interest rate.

## Syntax

PMT(rate,nper,pv,type)

Rate	is the interest rate for the loan, consistent with the number of payment periods. For
	example if the annual interest rate is 9% (0.09) and the payments are made monthly,
	then <b>Rate</b> should be 0.09/12.

- *Nper* total number of payments for the loan.
- *Pv* present value of the loan, also known as the principal.
- *Type* indicates when payments are due. Set to 0 if payments are due at the end of each period; 1 if payments are due at the beginning of each period.

This function is identical to Excel's PMT function with a future value of 0.

#### Examples:

■ Payment vs. Number of Payment Periods:



Payments on \$200,000 Loan

Payment vs. Interest Rate:



Payments on \$200000 Loan

# See also:

**Functions** 

## 6.3.40 RANDBETWEEN function

Returns a uniformly-distributed random number between limits you specify. RANDBETWEEN does not truncate the value to an integer (as the <u>RND</u> function does).

#### Syntax

RANDBETWEEN(bottom,top)

- *bottom* is the smallest number RANDBETWEEN will return.
- top is the largest number RANDBETWEEN will return.

See also: <u>NORMRAND function</u> <u>RND function</u>

#### **Functions**

### 6.3.41 RND function

Returns a random whole number between 0 and the input number. Argument must be  $\geq$  1, else you will get an "Invalid argument" error. To get a series of 5000 random numbers between 0 and 1, use RND (5000)/5000. Alternatively, use RANDBETWEEN(0,1)

See also:

NORMRAND function RANDBETWEEN function Functions

### 6.3.42 ROUND function

Rounds a number to a specified number of digits.

#### Syntax

ROUND(number,num_digits)

*number* is the number you want to round.

*num_digits* specifies the number of digits to which you want to round *number*.

If *num_digits* is greater than 0 (zero), then *number* is rounded to the specified number of decimal places.

If *num_digits* is 0, then *number* is rounded to the nearest integer.

If *num_digits* is less than 0, then *number* is rounded to the left of the decimal point.

ROUND(2.15, 1) = 2.2 ROUND(2.149, 1) = 2.1 ROUND(-1.475, 2) = -1.48 ROUND(21.5, -1) = 20

See also: <u>MROUND</u> Functions

### 6.3.43 SIGN function

Determines the sign of a number. Returns 1 if the number is positive, zero (0) if the number is 0, and -1 if the number is negative.

See also: Functions

### 6.3.44 SIN function

Returns the sine of the given angle. The angle may be in radians or degrees, controlled by the Angles options. With "radians" selected, sin(pi/2) returns 1.0



See also: ASIN function COS function Functions

# 6.3.45 SINH function

Returns the hyperbolic sine of the number.

$$sinh(x) = \frac{e^{X} - e^{-X}}{2}$$

sinh(x)



See also: ASINH function COSH function Functions

## 6.3.46 SQRT function

Returns the positive square root. The result multiplied by itself is equal to the sqrt's argument.

See also: Functions

## 6.3.47 SUM function

Adds the result of the argument to the previous result. This function is useful for evaluating Taylor series

The SUM function differs from all other functions in 2 respects:

You are limited to 10 SUM functions in any equation (or pair of equations in the case of X=f(T), Y=g (T)).

The **Prevent divide by zero errors** feature of the  $\underline{Y=f(X)}$  command and similar commands must be ignored when SUM is used in an equation. Division by 0 or other math error will result in an error message rather than incrementing X by a small value and re-evaluating.

#### Example

A well-known (though very slow to converge) method for calculating is:

$$\pi = 4 \left( 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} \cdots \right)$$

or

$$\pi = 4 \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{2n-1}$$

which can be expressed in DPlot as:

4*sum(((-1)^(x+1))/(2*x-1))

The mathematician and astronomer <u>Madhava of Sangamagrama</u> found a series that converges much faster than the above:

$$\pi = \sqrt{12} \left( 1 - \frac{1}{3 \times 3} + \frac{1}{5 \times 3^2} - \frac{1}{7 \times 3^3} + \dots \right)$$

or

$$\pi = \sqrt{12} \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{(2n-1) \times 3^{n-1}}$$

which may be expressed in DPlot as  $sqrt(12)*sum(((-1)^{(x+1)})/((2*x-1)*3^{(x-1)}))$ 

The first 20 terms of both series result in:



# 6.3.48 TAN function

Returns the tangent of the given angle. The angle may be in radians or degrees, controlled by the Angles options. With "radians" selected, tan(pi/4) returns 1.0.



ATAN function COS function SIN function Functions

# 6.3.49 TANH function

Returns the hyperbolic tangent of the number.


ATANH function Functions

#### 6.3.50 WEEKDAY function

Returns the day of the week corresponding to a date represented by an Excel 1900 date system serial number. The day is given as an integer, ranging from 1 (Sunday) to 7 (Saturday).

#### Syntax

WEEKDAY(serial_number)

#### See also:

DATE function DAY function MONTH function WEEKNUM function YEAR function Functions

#### 6.3.51 WEEKNUM function

Returns the number of weeks since the start of the year for a date represented by an Excel 1900 date system serial number.

#### Syntax

WEEKNUM(serial_number,return_type)

return_type is a number that determines on which day the week begins. 1=week begins on Sunday. 2=Week begins on Monday.

#### See also:

DATE function DAY function MONTH function WEEKDAY function YEAR function Functions

### 6.3.52 XOR function

Returns the result of the logical exclusive OR operation on the corresponding bits of two arguments. Both arguments are first truncated to integers; the result will always be an integer value. In C syntax, the result of XOR(a,b) is equivalent to (floor(a) ^ floor(b)).

Example:



NOT function OR function

### 6.3.53 YEAR function

Returns the year of a date represented by an Excel 1900 date system serial number.

### Syntax

YEAR(serial_number)

#### See also:

DATE function DAY function MONTH function WEEKDAY function WEEKNUM function Functions

## 6.4 Error Messages

Following is a description of selected error messages displayed by DPlot, with suggested corrective action.

#### **Clipboard Errors**

Unable to open Windows clipboard. OR

Unable to access clipboard ... try again later.

Either 1) another application is currently accessing the Clipboard (unlikely), or 2) another application failed to release control of the Clipboard after performing a cut/paste operation (more likely). Try again after a short delay. If this error message is repeated the only solution is to exit and restart Windows.

Data was successfully transferred, but the maximum number of points was exceeded. DPlot was unable to copy all of the data from the clipboard. You may use the File:Resize Arrays command and try again.

DPlot reads data from the Clipboard in one pass (for speed), so that it is not known how many data points are on the Clipboard until the points are actually transferred.

#### Memory Allocation

Insufficient memory for temporary storage.

Many DPlot functions require temporary storage for working variables. DPlot attempts to dynamically allocate memory for these temporary variables, and frees this memory when the function is complete. Under low memory conditions this procedure may fail. Close other active applications, if any, and try again. If no other applications are open, or if closing the applications does not solve the problem, set the array sizes for all current plots to the smallest possible values (maximum number of curves = current number of curves, maximum number of points = maximum number of points in any existing curve) using the Resize Arrays command on the File menu.

There is not sufficient disk space to temporarily store the existing data while resizing the X, Y arrays. See 'Error Messages' in Help.

It is possible that the addition of new data to an existing plot (either by reading a file or generating new data) would exceed the current array sizes (shown on the status bar and set with the Resize Arrays command). In this case DPlot attempts to first save the existing data to a temporary file, destroy the arrays, re-allocate the arrays with the new required size, and read the existing data back into memory from the temporary file. If there is not sufficient disk space to store the existing data this operation will fail. However, unlike the condition causing the following error message, DPlot does successfully restore the existing data. DPlot uses the drive set with the TEMP environment variable for the location of temporary files. If the TEMP environment variable is not set, DPlot uses the drive where Windows is located. Set the TEMP environment variable to the drive with the most free disk space (hopefully corresponding to your fastest drive).

#### Unable to restore arrays. Closing DPlot.

When attempting to resize the arrays as described above, if DPlot cannot allocate the arrays with the new required size, it attempts to restore the previous array sizes. Failure of this operation generally indicates serious memory. It is possible that a second active application has consumed some amount of memory between the times that DPlot destroys the original arrays and attempts to restore these arrays. Whatever the cause, if this condition occurs DPlot will terminate.

#### Scaling Errors

Manually entered window coordinates are not consistent with logarithmic scaling.

One or more of the plot extents set with the <u>Extents/Intervals/Size</u> command are less than or equal to zero. Change the plot extents using the Extents/Intervals/Size command on the Options menu.

Manually entered window coordinates are not consistent with a probability scale.

One or more of the plot extents set with the <u>Extents/Intervals/Size</u> command are either less than or equal to 0, or greater than or equal to 100. Change the plot extents using the Extents/Intervals/Size command on the Options menu.

Acceleration values exceed the limits of a single precision real number. A tripartite grid is not appropriate.

OR

Deflection values exceed the limits of a single precision real number. A tripartite grid is not appropriate.

Though the frequency (X) and velocity (Y) values are valid for a logarithmic scale, the associated acceleration or displacement values fall outside the limits of a single precision (4 byte) real number (approximately 1.2E-38 to 3.4E38).

No range for x values. (Xmin = Xmax) OR

No range for y values. (Ymin = Ymax)

These errors are generally associated with attempting to plot a single data point. At least two data points are required.

## 6.5 Known Problems

#### **Metafiles**

#### General

In general, metafiles produced by DPlot and copied to the Clipboard may be pasted into any application that supports the metafile format. For example, DPlot-produced metafiles may be pasted into any of the Microsoft Office products, usually without any significant problems. DPlot-produced metafiles will always work in applications that simply play back the metafile in its entirety. On the other hand, graphics-oriented applications that manipulate individual metafile records (MS PowerPoint and Corel Draw, for example) are more likely to mishandle portions of the metafile. If this is the case with your target application, please <u>contact us</u> with specific information on the problem (see the <u>Technical Support</u> topic.) Until the problem is resolved, you can **probably** get around the problem by saving the plot to disk as a metafile (using the <u>Save As</u> command on the File menu) and importing the metafile into the target application using the appropriate menu commands. In MS PowerPoint, for example, you can import a metafile using the Import Picture menu command.

#### Text in metafiles overlaps the surrounding border

This problem is due to a limitation on the sizes of available fonts. When a metafile is played back in a target application (e.g. a word processor or presentation graphics application), the font sizes originally chosen are scaled to the size of the picture, as is the rectangle surrounding this text. However, while the rectangle can be scaled to any size, the corresponding font size may not be available (Windows selects the closest size available). For best results, select the desired size of the metafile using the <u>Copy Dimensions</u> command or <u>Extents/Intervals/Size</u> command **before** copying a plot to the Clipboard or saving as a metafile. This is in general much less of a problem when working with enhanced metafiles than standard Windows metafiles, assuming the target application supports that format.

#### **Microsoft Office 97-specific problems**

DPlot copies both enhanced metafiles and standard Windows metafiles to the Clipboard for most

2D plots. The default format when pasting into a target application will usually be an enhanced metafile, assuming that the target application supports this format. However, Office 97 products in some cases do not interpret enhanced metafiles correctly, and very often cut off the right and/or bottom side of a plot. This is not a DPlot issue, but a problem with older Office products. If you are using Windows 95 and Office 97 or older Office products and run into clipping problems when pasting DPlot enhanced metafiles, use the target application's *Paste Special* command instead of *Paste* and choose the *Picture* format rather than *Picture* (*Enhanced Metafile*).

#### Microsoft PowerPoint

In many versions of PowerPoint, the default format when using *Paste* will be a standard metafile rather than an enhanced metafile. With standard metafiles you may notice text placement problems, particularly if you make use of DPlot's subscript and/or superscript features. To paste an enhanced metafile rather than a standard metafile, use *Paste Special* rather than *Paste*, and select *Picture (Enhanced Metafile)*.

#### Printing

#### Grid lines do not print

One or more grid lines may not be printed if your printer must simulate color using patterns of dots and the line width selected for the grid lines (using the <u>Symbol/Line Styles</u> command on the Options menu) is so small that the dithered pattern used by the printer driver produces all white dots. The grid line width is set to 0.01 inches (0.254 mm, 0.72 points) by default. If one or more grid lines are not shown on your printed output, try increasing the grid line width to a larger value.

#### Rotated text with HP LaserJet 4/4M

The HP LaserJet 4/4M printer driver, version 31.1.1.8 or earlier, does not place rotated text correctly. The LaserJet 4/4M printer driver, version 31.V1.50, places rotated text correctly but in some cases drops characters from rotated character strings.

To fix either of these problems:

Use the HP LaserJet 3 driver distributed with Windows. Although this solution will fix the problem, output is limited to 300 dpi.

OR

Use the HP LaserJet 4/4M PostScript printer driver. This driver is usually included on 3-1/2" disk with the printer.

OR

Obtain an updated driver (L4WIN3.EXE) from Hewlett Packard. Driver updates are available on the World Wide Web at <u>http://www.hp.com</u> or from the Hewlett Packard Download Service at (208) 344-1691. If this does not fix the problem, it may be necessary to select "Raster" Graphics Mode and check "Print TrueType as Graphics" in the Print Setup dialog box. Note, however, that these print options result in a much larger printer spool file, and printing speed is considerably slower.

#### Colors

Although each document may have a unique color palette, on 256-color displays it is the active document's palette that will be used on all visible documents. The only alternative to this behavior is to stop using the Windows palette management and show dithered colors to simulate your color selections, which is, at best, ugly. This is not a problem on 16-, 24-, or 32-bits/pixel displays.

#### Limited Help functionality on older Windows versions

The DPlot Help file and the Help files for all plugin modules distributed with DPlot use HTML Help. HTML Help is known to have problems on Windows 2000 and earlier versions: attempting to display a HTML Help file using the HHCTRL.OCX function HtmlHelp results in a crash. To get around this problem, on Windows 2000 and earlier versions DPlot launches the Help file in a separate process, which works on all Windows versions. The main limitation of doing this is that context-sensitive help in dialog boxes (popup windows displayed after clicking the question mark button in the title bar) is disabled.

#### Display Problems

#### Fill Between Curves

The built-in hatch patterns (horizontal lines, vertical lines, etc.) are not drawn correctly on systems equipped with ATI Mach32 (and possibly other) graphics adapters. The user-defined pattern, however, does work properly.

Problems and/or suspected bugs not referenced in this topic should be reported to the author as described in the <u>Technical Support</u> topic.

## 6.6 Acknowledgements

DPlot makes use of several publicly-available modules that enhance its functionality:

#### RunicSoft Function Parser, by Andreas Hartl

A heavily modified version of this DLL is used by the  $\underline{Y=f(X)}$  command and similar commands to generate and/or modify data with a user-defined function.

#### ZLIB Compression Library

ZLIB is used to save and read <u>compressed DPlot files</u>. Compression is lossless. Files compressed using this method will have a compression ratio from 2.5:1 to 4:1. Larger files may achieve 12:1 or higher.

#### FreeImage Image Library

FreeImage is a toolkit for for developers interested in supporting a variety of image formats. DPlot uses FreeImage to save BMP, GIF, PNG, JPG, and TIF file formats, and for importing background images.



## 7 Technical Support

## 7.1 Contact information

Your comments, suggestions, and questions are always welcome. We can be reached in a number of ways:

Email:	support@dplot.com
Online forums:	http://www.dplot.com/forums/index.php
Phone:	601-629-8607
Postal mail:	HydeSoft Computing LLC 110 Roseland Drive Vicksburg, MS 39180 USA

#### **Bug reports**

Before reporting errors, please see the description of known problems and possible solutions in the **Known Problems** topic.

Please include DPlot's full version number (select the **About DPlot** command on the Help menu) in any correspondence.

#### Updates to DPlot

Updates to the evaluation version of DPlot are available from the DPlot home page at <a href="http://www.dplot.com">http://www.dplot.com</a>. Licensed users may opt to receive automatic notifications when new versions are available. Updates are free up to 2 years after date of purchase. If you are a licensed DPlot user not currently on our mailing list and wish to subscribe to the list, just <a href="http://www.dplot.let.us.know">let.us.know</a>. (Be sure to include your username and registration number from the About box.) Alternatively, licensed users may select the Check for Updates command on the Help menu to download and install new versions. If you purchased a DPlot license more than two years ago you may purchase an upgrade at a 2/3 discount off the normal price. To take advantage of this offer use the Check for Updates command on the Help menu or contact us by e-mail here.

#### HydeSoft Computing Information

For purchasing agents who need this information:

DUNS number:	962092305
US EIN:	30-0359350
US Form W-9 (PDF):	http://www.dplot.com/w9_hydesoft.pdf
SBA ID:	P1572656
CCR CAGE:	6HT29

## 7.2 Notes for DPlot Jr and DPlot Viewer users

DPlot Jr and DPlot Viewer are scaled-down versions of DPlot intended for different audiences. DPlot Jr is designed for use by programmers that want a quick method for displaying graphical output of their own applications. DPlot Viewer is intended for use by those who want end users to be able to view graphs produced by DPlot or DPlot Jr. Unlike the full version of DPlot, DPlot Jr and DPlot Viewer may be redistributed to end users without royalty payments or other restrictions. Input to DPlot Jr is restricted to DDE or calls to functions in <u>DPLOTLIB.DLL</u>; the end user **cannot open a file** or paste data from the clipboard. DPlot Jr. Both DPlot Jr and DPlot Viewer will not **create** a graph; it only displays previously-existing graphs produced by DPlot or DPlot Jr. Both DPlot Jr and DPlot Viewer lack many of the editing and data generation features of the full version. For more information go to the <u>DPlot web site</u>.

### **Feature Limitations**

All features and functions described in this Help file are available in the full version of DPlot. In general, the descriptions of menu commands, macro commands, and other features that **are** available in the full version of DPlot but not in DPlot Jr or DPlot Viewer will include a key describing the limitations:

√JR	indicates that the feature is supported by DPlot Jr.
¥JR	indicates that the feature is <b>not</b> supported by DPlot Jr.
✓Viewer	indicates that the feature is supported by DPlot Viewer.
×Viewer	indicates that the feature is $\ensuremath{\text{not}}$ supported by DPlot Viewer.

### Distribution

You are authorized to make as many copies of DPlot Jr and/or DPlot Viewer and accompanying documentation as you wish; give exact copies of the program(s) to anyone; and distribute these program(s) and documentation in their unmodified form via electronic means or via CD or other media. There is no charge for any of these actions.

When building your distribution, these files are required:

DPLOTJR.EXE or DPLOTVU.EXE	The main executable
DPLOT.CHM	This Help file
CONTEXT.CHM	Context-sensitive Help file for dialog boxes
UTILS.DLL	Utilities library
FPARSER.DLL	Equation parser
FREEIMAGE.DLL	Graphics export utilities; used by DPlot Jr to save bitmap images.
TOOLS.DLL	Toolbar images
ZLIB1.DLL	Compression library using the ZLIB algorithm. Used to compress/ decompress DPlot files on the fly.
DPMAC.EXE (DPlot Viewer only)	Macro editor

All of these files are located in the folder where you initially installed DPlot Jr or DPlot Viewer. These unmodified files **must** be included in your distribution; the programs will not work correctly without them. Each of these files should be installed to the same folder on the end user's system. It is **not** necessary to install these files to your application's folder. Generally a subfolder below your application folder is best.

If your application sends data to DPlot Jr via DPLOTLIB.DLL, of course you will need to install this

DLL as well. For best results install DPLOTLIB.DLL to your application's folder, **not** the Windows System folder. Although the System folder will work, you cannot guarantee that some **other** application's installation will not overwrite the version you distribute with an older version.

Your application's installation procedure should make one change to the end user's registry that tells DPLOTLIB.DLL (and possibly your own application) where to find DPlot Jr:

HKEY_CURRENT_USER Software DPLOTJR shell open command Data Type=REG_SZ Value = path to DPlot Jr, including the filename

For example, the following is the section of the setup script for <u>Inno Setup</u> that modifies the end user's registry:

#### [Registry]

Root: HKCU; Subkey: "Software\DPLOTJR\shell\open\command"; ValueType: string; ValueData: "{app}\dplotjr.exe"; Flags: uninsdeletekey

DPlot Jr will itself update the registry with its own location every time it runs. The above change ensures that your application and/or DPLOTLIB can find DPlot Jr before it ever runs on the end user's system. Note that the path should be surrounded by double quotation marks (handled automatically by Inno Setup). Otherwise if the path contains a space (as in c:\Program Files), your program's attempts to launch dplotir.exe will likely fail.

When testing your setup program you should ensure that the correct path was written to the registry. Using regedit after running your setup program you should now have

HKEY_CURRENT_USER\Software\DPLOTJR\shell\open\command with the Name left at (Default), the Type set to REG_SZ, and the path to dplotjr.exe, surrounded by double quotation marks, under Data.



## 8 Transferring data

## 8.1 Using the DPIot Interface Add-In for Microsoft Excel

If you elected to install the DPlot Interface add-in when you installed DPlot, you should see a "DPlot" menu entry in Microsoft Excel 2003 and previous versions:

	Microsoft Excel - Book1											
:2)	Eile	<u>E</u> dit	⊻iew	Insert	F <u>o</u> rma	t <u>T</u> ools	Data	<u>W</u> indow	DPI	ot <u>H</u> elp		
	2		a 1	6 🖪	ABC 1	1 8 1	b 🖪	+ 🛷 🖻		<u>X</u> YXY		
	A1		-	fx						XYYY K		
	100	4	В		С	D		E		OneCurvePer <u>R</u> ow		
1		<u> </u>								XY <u>Z</u> Surface		
2										XYZ <u>S</u> catter		
4										Z <u>G</u> rid		
5									-	<u>O</u> neD —		
<u>ь</u> 7									-	Append to active document		
8										Heln		
9										Tohu		

In Excel 2007, if you select the Add-Ins tab then you should see a DPlot entry at the left of the screen under the Home tab:

0	n 🖬 🔊 -	(≈						examples.	xls [Compat	tibi
C	Home	Insert	Page Layout	Formulas	Data	Review	View	Developer	Add-Ins	1
	DPlot -								2	
Mei	nu Commands									
	B1	<del>•</del> (	● ƒ _× X							
		А		В	С	D	E	F	G	
1	XYXY test Similar to X form X1,Y1, Each pair o the same si	(YYY, but o X2,Y2, of columns	data is in the must be within	x	¥1		¥2		¥3	
2	and sume s	ore of the		5.000	0.000	4.000	0.000	3.000	0.000	
3				1 999	0.087	3 999	0.087	2 999	0.087	

In either case, if you do not see something similar to the above then either 1) you did not install the optional Add-In when you installed DPlot, 2) the Add-In actually **was** installed but a security setting in Excel has hidden it, or 3) a user with an account different than your own (most likely an administrator) installed DPlot for you. If you did **not** install the Add-In you can do so simply by re-installing DPlot to the same folder where it currently exists (the default is c:\Program Files\DPlot), this time making sure to check the "MS Excel Add-In" box. You will need to exit DPlot before re-installing.

If you did install the Add-In but do not see a DPlot menu in Excel:

#### In Excel 2003 and previous verions

Select "Add-Ins" on the Tools menu:

🗷 M	Microsoft Excel - Book1											
:2	Eile	<u>E</u> dit	⊻iew	Inse	rt F <u>o</u>	rmat	Tool	ls	<u>D</u> ata	Window		
10	2 I			8	AB	1		М	acro	• 7		
-	A1		-	f:	è			A	dd-Ins	· N		
	A		В		С			Q	ustomize			
1		ļ					Options					
2							*					
4												

There should be a "DPlot Interface" box:



Check that box, then click OK. If there is **not** a "DPlot Interface" box, then most likely DPlot was installed by a different user than you, probably an administrator because you have a limited-privileges user account. In that case click the "Browse" button and navigate to the folder where DPlot was installed (default location=c:\Program Files\DPlot). Select the file dplotlib.xla then click OK.

#### In Excel 2007 and later

Click the Office button in the upper left corner:

	) 🗔 🤊	~ 🕅 -	₹							
e	Home	Ins	ert	Page Layout	Formula	as Data				
Pa	Office Butto	DD 	Click and t do w	here to open o see everyth ith your docu	, save, or pri ing else you iment.	nt, can				
0										
1	Press F1 for more help.									
2		0	0	0	0	0				
2		1	1	າ	2	л				

On the resulting screen click "Excel Options" at the bottom of the window.

Click "Add-Ins" in the left pane.

At the bottom of the next screen, ensure that the "Manage" selection is "Excel Add-Ins" then click "Go"

The next dialog is identical to that for Excel 2003 shown above. Check the "DPlot Interface" box (or follow the instructions above if that box is not present), then click OK.

If you properly installed the Add-In but now when starting Excel 2003 or previous versions you see an error message complaining about **Run-time error '438**', the most likely cause is some previous operation that has, for whatever reason, caused Excel's toolbar to temporarily lose its mind. You can correct this problem with the following steps:

- Start Excel. Select Tools>Customize. Click the Toolbars tab.
- Near the end of the Toolbars list, find Worksheet Menu Bar. Select that entry so that it is highlighted.
- Click the Reset button. When asked whether you really want to do this, click OK.
- Click the **Close** button, then exit and restart Excel.
- If you previously deactivated the DPlot Add-In to get rid of this error message, select Tools>Addins, check the DPlot Interface box, then click OK.

#### In Excel 2013 and later

If you followed the instructions above under "In Excel 2007 and later" and either the Add-Ins tab is not shown or the DPlot menu is not displayed when clicking the Add-Ins tab, it is most likely because of new functionality in Excel 2013 that gives every workbook a unique ribbon, which the Add-In is not currently set up to handle. Rather it **might** handle it, but not in all cases. In this case you will need to load the DPlot Add-In separately as a workbook. First, unload the Add-In by selecting Excel Options>AddIns>Manage as described above, then uncheck the "DPlot Interface" entry and click OK. Next open whatever workbook you wish to work on if it is not already open, and finally open the Add-In's source workbook DPLOTLIB.XLS in the DPlot folder below your "My Documents" folder. You **should** now see an Add-Ins tab on the ribbon and a DPlot menu when you click "Add-Ins". This situation will be resolved in a future DPlot release.

To use the Add-In, start Microsoft Excel and open the worksheet containing the data you'd like to plot in DPlot. If you are unfamiliar with the Add-In you will find the example Excel spreadsheet EXAMPLES. XLS useful. This sheet includes examples for all of the functions in the Add-In. EXAMPLES.XLS is installed to the DPlot folder below your "My Documents" folder.

The Add-In contains functions that allow you to pass data from Excel to DPlot, producing any type of plot that DPlot currently supports.

#### Key Add-In features:

- For all functions other than ZGrid (see below), you may use multiple selections if your data columns are not adjacent. This differs from using Copy/Paste, which results in copying ALL values within the extents of all selections.
- Data is passed as floating point numbers and internally is independent of the formatting. With Copy/ Paste, data is passed as text, exactly as it appears in your worksheet. If you only show 2 decimal places then the data is truncated to 2 decimal places in DPlot.
- Blank cells are interpreted as you'd expect.
- All functions other than **OneD** attempt to use the same formatting in DPlot that is used in Excel (dates, for example).

#### Add-In Functions

#### <u>XYXY</u>

Select alternating X,Y columns and produce an XY plot. Column selections may have differing

numbers of rows.

#### <u>XYYY</u>

Same as above, but X is taken from the first selected column and all subsequently-selected columns are Y values for separate curves. If only one column is selected, it is assumed to be Y values and X is generated, starting at 0 and incremented by 1. Y columns may have blank cells.

#### X,Y,Label

Creates 1 curve on an XY plot with X is taken from the first selected column, Y taken from the second selected column, and text labels in the third column. Label cells may be blank.

#### **Bar chart**

Creates a bar chart with non-numeric labels on the X axis and 1 or more groups of amplitudes.

#### Bar chart, data in rows

Same as Bar chart command, with data sets in the same row rather than the same column.

#### OneCurvePerRow

Similar to XYXY, but data is arranged by rows rather than columns, with alternating X and Y values in adjacent columns. Each row is used to produce one curve.

#### XYZSurface

Select an X, a Y, and a Z column (one or more selections). Produce a 3D surface plot.

#### **XYZScatter**

Select one or more groups of X,Y,Z columns and produce a 3D scatter plot, one data set per X,Y,Z group.

#### **ZGrid**

Select a table with X in the first row, Y in the first column, and the remainder of the table filled with Z values for the corresponding X and Y. Produces a 3D surface plot.

#### <u>OneD</u>

Select one or more columns of amplitudes and produce a box-and-whisker plot.

#### Editing the Add-In

The Visual Basic for Applications (VBA) routines in DPLOTLIB.XLA are protected against editing, mainly to avoid confusion with other VBA functions you might create. If you are comfortable with VBA and would like to experiment with the add-in's features:

- We strongly suggest that you first make a backup copy of the .XLA file. In Office 2000 and later versions, DPLOTLIB.XLA can be found in the \Microsoft\Addins folder below the Application Data folder. For earlier versions of Excel DPLOTLIB.XLA will be stored in the \Library folder below EXCEL.EXE. Save the backup copy of the Add-In to a different folder so that you won't become confused on Excel's Add-In's dialog.
- 2) Unload the Add-In: Select Tools>Add-Ins and uncheck "DPlot Interface".
- Select Tools>Macro>Security. On the Security Level tab, ensure that "Low" is checked. Don't be concerned about changing your security level; this is only a temporary change that will enable you to edit and run VBA functions.
- 4) Open the Add-In file (DPLOTLIB.XLA, see above for the location of this file) using File>Open.
- 5) Select Tools>Macro>Visual Basic Editor:

Tools Data Window	<u>H</u> e	elp D <u>P</u> lot	
Spelling F7	ся +		📶 📣 100% ·
<u>M</u> acro ►	•	<u>M</u> acros	Alt+F8
Add- <u>I</u> ns	•	Record New Macro	
*		Security	
	2	Visual Basic Editor	Alt+F11
		Microsoft Script Editor	Alt+Shift+F11

In the Project Explorer window, click on the + symbol next to DPlotLib (dplotlib.xla):



If you do not see the Project Explorer window, select *Project Explorer* on the *View* menu. After clicking on the + symbol next to **DPlotLib (dplotlib.xla)** you will be prompted for a password. Enter "dplot" in all lowercase letters (without the quotation marks).

- 6) The functions under *Microsoft Excel Objects (This Workbook)* simply create the DPlot menu and delete the menu if/when the add-in is unloaded. The meat of the add-in is under *Modules* in *DPlotData*. For more information on DPLOTLIB functions see the DPLOTLIB documentation at http://www.dplot.com/lib/index.htm
- After making any changes, save your work, exit the VBA editor, reset your security level if necessary, and re-load the add-in using Tools>Add-Ins.

#### **VBA Programming**

If you find that the Add-In commands almost, but not quite, meet your needs and you are reasonably comfortable with VBA, you can call the standard Add-In commands from your own VBA routine and then make changes to a plot, typically by sending command strings to DPlot. A couple of examples follow.

1) Data arranged in columns such that the normal Add-In commands can produce the desired plot from a selection:

```
Sub monthlysalesmacro()
Dim doc As Long
Dim ret As Long
Dim ret As Long
Dim Sel As Object
Dim row As Long
Dim rnge As String
Dim d As Double
Set Sel = Selection ' Preserve user's current selection
Range("D:D,I:I").Select ' dates in column D, cumulative monthly
sales in column I
Call XYXY
doc = DPlotGetActiveDocument()
```

```
If doc > 0 Then
     ' generate a new curve with amplitudes equal to the peak value
    for each month
    ret = DPlot_Command(doc, "[RunPlugin(""Monthly Peak"")][
    SymbolSize(-1,150)]")
    ' delete the original curve
    ret = DPlot_Command(doc, "[SelectCurve(1)][EditErase()]")
    ret = DPlot_Command(doc, "[DateFormat(""yyyy"")]")
    ret = DPlot_Command(doc, "[Title1(""My Sales"")][Title2("""")][
    NumberFormat(1,4)]")
    ' set the tick mark intervals to 365 days on X, $1000 on Y
    ret = DPlot_Command(doc, "[TickInterval(1,365,1000)]")
     ' set the axis extents: whole years on X, 0 to next highest $1000
    increment on Y
    ret = DPlot_Command(doc, "[ManualScale(" & Str$(DateSerial(2001,
    1, 1)) & ",0," & _
        DateSerial(year(Now) + 1, 1, 1) & ",=(CEIL($YMAX/1000)
    *1000))]")
     ' generate 12-month moving average using previous 12 months
    ret = DPlot_Command(doc, "[RunPlugin(""Moving
    Average"", ""1, 12, 0, 1"")]")
    Call DPlot Finish(doc)
  End If
  Sel.Select
                               ' restore previous selection
End Sub
```

A few notes on the above:

- a) To call the Add-In functions DPlot_Command or DPlot_Request you need the index of the currently active document (XYYY and other Add-In commands activate the plots that they create). That's where DPlotGetActiveDocument comes in. For a rather lengthy list of all available DPlot_Command commands, see <u>Sending data to DPlot from another application</u>.
- b) When finished with calling any DPlot functions in your macro, if you have called DPlot_Command or DPlot_Request, you should disconnect from the currently active DDE conversation. Otherwise the plot will work as expected, but certain mouse operations (e.g. displaying coordinates while moving your mouse across the plot) will not work as expected, as DPlot will think there is still an active DDE conversation and responding to those actions is not appropriate. Disconnecting from a conversation can be accomplished simply with: Call DPlot_Finish(doc)
- c) In the VBA editor select Tools>References and check dplotlib. If you do **not** do this you will get a "Sub or Function not defined" error from Excel.
- Data produced by another VBA module or data source or not conveniently arranged for the Add-In functions:

```
Sub Test()
Dim d As DPLOT ' DPLOT structure defined in the Add-In
Dim doc As Long
Dim ret As Long
Dim x(0 To 3) As Double
Dim y(0 To 3) As Double
x(0) = 1
x(1) = 2
x(2) = 3
```

```
x(3) = 4
   y(0) = 1
   y(1) = 2
   y(2) = 3
   y(3) = 4
   d.Version = DPLOT_DDE_VERSION
   d.DataFormat = DATA_XYXY
   d.MaxCurves = 1
                             ' Must be >= number of curves
                            ' Must be >= maximum number of points for all curve
   d.MaxPoints = 4
   d.NumCurves = 1
                            ' Actual number of curves
   d.NP(0) = 4
                             ' Number of points in first curve
   d.LineType(0) = 1
   d.ScaleCode = SCALE_LINEARX_LINEARY
   DPlot_Plot8 d, x, y, ""
   doc = DPlotGetActiveDocument()
    If doc > 0 Then
       ret = DPlot_Command(doc, "[AppShow()]")
       Call DPlot_Finish(doc)
    End If
End Sub
```

The same notes from the previous example apply here.

#### 8.1.1 XYXY Add-In command

Select alternating X,Y columns and produce an XY plot. Column selections may have differing numbers of rows.

You may use multiple selections (by use of the Ctrl key). The first column selected is taken to be the X axis values for the first curve; the second column selected is used for the Y values of the first curve, etc. So if, for example, your data is arranged with Y values in column A and X values in column B, you should first select column B, then use the Ctrl key to append column A to the selection.

If the data (including column headings, if any) you'd like to plot begins in row 1, you can select entire columns by clicking on the column headings (A, B, C, etc.). The XYXY function assumes that data for any curve stops at the first blank X cell.

#### Initial plot settings

#### Symbols and lines

Intially, XY plots produced by the Add-In will consist of line segments between each data point, with no symbols. You can easily switch to a scatter plot by:

- 1) Select Scatter Plot on the Options menu. For any curves that do **not** have a symbol selected, filled circles are used. Line type is set to "None". Or
- 2) Right-click on a curve, select "Symbol style", then repeat for the "Line style", setting the style to "None". Repeat for all curves.

If you are working with more than a couple of curves you may prefer to use the Symbol/Line Styles command on the Options menu.

#### Scaling (linear, logarithmic, polar, etc.)

Initially, XY plots generated by the Add-In will use linear scaling on both the X and Y axes. To change to a different format (logarithmic scaling, polar plot, etc.), right-click on the plot and select the scaling you'd prefer.

#### Example:



## 8.1.2 XYYY Add-In command

Similar to the XYXY command, but X is taken from the first selected column and all subsequentlyselected columns are Y values for separate curves. If only one column is selected, it is assumed to be Y values and X is generated, starting at 0 and incremented by 1. Y columns may have blank cells.

If the X values for your data are **not** found in the leftmost column, you can use multiple selections to force the expected behavior: select the X data, then use the Ctrl key to append Y columns to the selection.

Your selection may include blank cells, which will be ignored.

If the data (including column headings, if any) you'd like to plot begins in row 1, you can select entire columns by clicking on the column headings (A, B, C, etc.).

#### Initial plot settings

Intially, XY plots produced by the Add-In will consist of line segments between each data point, with no symbols. You can easily switch to a scatter plot by:

- 1) Select Scatter Plot on the Options menu. For any curves that do **not** have a symbol selected, filled circles are used. Line type is set to "None". Or
- 2) Right-click on a curve, select "Symbol style", then repeat for the "Line style", setting the style to "None". Repeat for all curves.

If you are working with more than a couple of curves you may prefer to use the Symbol/Line Styles command on the Options menu.

#### Scaling (linear, logarithmic, polar, etc.)

Initially, XY plots generated by the Add-In will use linear scaling on both the X and Y axes. To change to a different format (logarithmic scaling, polar plot, etc.), right-click on the plot and select the scaling you'd prefer:

### Example:



#### 8.1.3 X,Y,Label command

Creates one curve with optional labels at one or more data points.

X is taken from the first selected column, Y from the second selected column, and labels from the third selected column. If your columns are not in X,Y,label order, you can make multiple selections by use of the Ctrl key. Select the X data first, press Ctrl and select the Y data, then press Ctrl and select the labels. Each selection should begin in the same row.

If the data (including column headings, if any) you'd like to plot begins in row 1, you can select entire columns by clicking on the column headings (A, B, C, etc.). The XYXY function assumes that data for any curve stops at the first blank X cell.

#### Label placeholders

To embed the amplitude of a data point in a label, type \$Y in the Label field. To embed the abscissa of a data point in a label, type \$X. When the label is drawn the corresponding numeric values will be substituted for these control codes. For example, a label of "\$X sec, \$Y psi" at the point (5.5,100) will appear as "5.5 sec, 100 psi".

#### Initial plot settings

Intially, XY plots produced by the Add-In will consist of line segments between each data point, with no symbols. You can easily switch to a scatter plot by:

- 1) Select Scatter Plot on the Options menu. For any curves that do **not** have a symbol selected, filled circles are used. Line type is set to "None". Or
- 2) Right-click on a curve, select "Symbol style", then repeat for the "Line style", setting the style to "None". Repeat for all curves.

If you are working with more than a couple of curves you may prefer to use the Symbol/Line Styles command on the Options menu.

#### Scaling (linear, logarithmic, polar, etc.)

Initially, XY plots generated by the Add-In will use linear scaling on both the X and Y axes. To change to a different format (logarithmic scaling, polar plot, etc.), right-click on the plot and select the scaling you'd prefer.

#### Example:



#### 8.1.4 Bar chart command

Creates a bar chart with non-numeric labels on the X axis with one or more groups of amplitudes. The X axis labels should be in the first column selected. Amplitudes should be in the second and subsequent columns. To select multiple non-adjacent columns, press the Ctrl key while making a selection.

If more than 10 labels are selected, the Add-In will instruct DPlot to orient the labels vertically. To change the label orientation within DPlot, select Options>Number Formats.

#### Example:

	Fruit	Piggly Wiggly	Jitney Jungle	Albertson's	
	Apples	474	1354	425	
	Bananas	1960	1201	1956	
	Oranges	1208	1042	1330	
	Grapes	1729	542	623	
٦	Limes	679	720	1425	



#### Initial plot settings

Initially the bar chart will have filled bars. If multiple data sets are graphed then the bars will be drawn side-by-side. You can quickly change bar chart settings (stacked bars, amplitude labels, bar width, etc.) within DPlot by right-clicking on the chart and selecting "Bar Chart Options".

#### 8.1.5 Bar chart, data in rows command

This command is very similar to the Bar chart command, but expects the data to be transposed: each column is for the same X (bar group), each row is for a different data set. If more than 1 row is selected, then the first row **must** contain either X values or non-numeric labels (as in the example below) for the X axis. If only 1 row is selected, X values will automatically be generated (0,1,2,...).

#### Example:

Fruit	Apples	Bananas	Oranges	Grapes	Limes
Piggly Wiggly	536	953	1875	126	977
Jitney Jungle	1101	853	1403	1482	1546
Albertson's	800	216	1830	85	1830



#### Initial plot settings

Initially the bar chart will have filled bars with amplitude labels drawn above each bar. If multiple data sets are graphed then the bars will be drawn side-by-side (as opposed to stacked). You can quickly change bar chart settings (stacked bars, amplitude labels, bar width, etc.) within DPlot by right-clicking on the chart and selecting "Bar Chart Options".

## 8.1.6 OneCurvePerRow Add-In command

Similar to XYXY, but data is arranged by rows rather than columns, with alternating X and Y values in adjacent columns. Each row is used to produce one curve:

Curve1:	X(1,1)	Y(1,1)	X(1,2)	Y(1,2)	X(1,3)	Y(1,3)	 X(1,N1)	Y(1,N1)
Curve2:	X(2,1)	Y(2,1)	X(2,2)	Y(2,2)	X(2,3)	Y(2,3)	 X(2,N2)	Y(2,N2)

#### Initial plot settings

#### Symbols and lines

Intially, XY plots produced by the Add-In will consist of line segments between each data point, with no symbols. You can easily switch to a scatter plot by:

- 1) Select Scatter Plot on the Options menu. For any curves that do **not** have a symbol selected, filled circles are used. Line type is set to "None". Or
- 2) Right-click on a curve, select "Symbol style", then repeat for the "Line style", setting the style to "None". Repeat for all curves.

If you are working with more than a couple of curves you may prefer to use the Symbol/Line Styles command on the Options menu.

#### Scaling (linear, logarithmic, polar, etc.)

Initially, XY plots generated by the Add-In will use linear scaling on both the X and Y axes. To change to a different format (logarithmic scaling, polar plot, etc.), right-click on the plot and select the scaling you'd prefer:

### 8.1.7 XYZSurface Add-In command

Select an X, a Y, and a Z column (one or more selections), and optionally a fourth column containing labels for one or more data points. Produce a 3D surface plot. NOTE: Labels are **only** drawn in 2D views of the surface, not in 3D views.

#### Label placeholders

To embed the X, Y, and/or Z values of the corresponding data point in a label, use the placeholders X, Y, and Z, respectively. When the label is drawn the corresponding numeric values will be substituted for these control codes. For example, a label of "X=X, Y=Y,Z=Z" at the point (1,2,3) will appear as "X=1,Y=2,Z=3".

#### Initial plot settings:

Initially the plot will be shown in 2D with contour lines. To show in 3D, draw shaded bands for each contour interval rather than contour lines, change the contour interval or the colors used, right-click outside the plot and select "Contour Options", or select the Contour Options command on the Options menu.

Surface plots in DPlot are of a triangular mesh that is generated from your 3D input points. There is not enough information in a simple list of 3D points to determine where the edges of the surface should be, or for that matter to determine whether your surface should have any interior holes. DPlot always creates a convex triangular mesh from X,Y,Z input, which for many applications is exactly what you want. It is of course possible that you will not want a convex mesh. In this case DPlot will have created triangles outside the desired limits of your plot. You can create a concave mesh by telling DPlot where the boundaries of your surface are with the Define Boundary command on the Options menu. For more information on this command search the DPlot Help file for "Define boundary".

#### Example:



### 8.1.8 XYZScatter Add-In command

Select one or more groups of X,Y,Z columns and produce a 3D scatter plot, one data set per X,Y,Z group.

#### Initial plot settings:

If more than one data set (3 columns) is selected, then the column heading for the Z values, if present, will be used as a legend.

If more than one data set is selected, symbols are colored according to which data set they belong to. If one data set is selected, symbols are colored by Z value. These options may be changed by right-clicking on the plot and selecting "Scatter Plot Options".

All symbols will initially be drawn as circles. Symbol type and size may be changed by right-clicking on the plot and selecting "Symbol Styles".

#### Example:



Second example, graphing 3D parametric equations:



## 8.1.9 ZGrid Add-In command

Select a table with X in the first row, Y in the first column, and the remainder of the table filled with Z values for the corresponding X and Y. Produces a 3D surface plot.

#### Initial plot settings:

Initially the plot will be shown in 2D with contour lines. To show in 3D, draw shaded bands for each contour interval rather than contour lines, change the contour interval or the colors used, right-click outside the plot and select "Contour Options", or select the Contour Options command on the Options menu.

#### Example:

	A	В	С	D	E	F	G	Н	1	J	K	L	М	Ν	0	Р	Q	R	S	Т	U	V	W	Х	Y	Ζ
1	ZGrid test																									
2	Select B6:	AG37	(or of	cours	e crea	ate yo	ur ow	n tabl	e), the	en sele	ect ZG	irid or	n the l	)Plot I	menu.											
3	First row i	s X va	lues,	first c	olumn	is Y.	The re	emain	der o	f the ta	able i	s Z va	lues a	t the o	corres	pondi	ng X a	and Y								
4	It is not ne	cessa	ry for	the ro	ows ar	nd col	umns	to be	even	y-spac	ced, tl	nough	ı they	are in	this e	examp	ole.									
5																										
6		Y/X	-3.00	-2.80	-2.60	-2.40	-2.20	-2.00	-1.80	-1.60	-1.40	-1.20	-1.00	-0.80	-0.60	-0.40	-0.20	0.00	0.20	0.40	0.60	0.80	1.00	1.20	1.40	1.60
7		-3.00	0.14	0.33	0.51	0.67	0.80	0.90	0.96	0.99	0.98	0.92	0.83	0.71	0.56	0.39	0.20	0.00	-0.20	-0.39	-0.56	-0.71	-0.83	-0.92	-0.98	-0.99
8		-2.80	0.13	0.32	0.49	0.64	0.76	0.86	0.92	0.94	0.93	0.88	0.79	0.68	0.53	0.37	0.19	0.00	-0.19	-0.37	-0.53	-0.68	-0.79	-0.88	-0.93	-0.94
9		-2.60	0.12	0.29	0.44	0.58	0.69	0.78	0.83	0.86	0.84	0.80	0.72	0.61	0.48	0.33	0.17	0.00	-0.17	-0.33	-0.48	-0.61	-0.72	-0.80	-0.84	-0.86
10		-2.40	0.10	0.25	0.38	0.50	0.60	0.67	0.72	0.74	0.73	0.69	0.62	0.53	0.42	0.29	0.15	0.00	-0.15	-0.29	-0.42	-0.53	-0.62	-0.69	-0.73	-0.74
11		-2.20	0.08	0.20	0.30	0.40	0.48	0.54	0.57	0.59	0.58	0.55	0.50	0.42	0.33	0.23	0.12	0.00	-0.12	-0.23	-0.33	-0.42	-0.50	-0.55	-0.58	-0.59
12		-2.00	0.06	0.14	0.21	0.28	0.34	0.38	9 K	DPlot	- [exar	nplesa	ds]	1.00			- 18		-					x	-0.41	-0.42
13		-1.80	0.03	0.08	0.12	0.15	0.18	0.21	0	) Eile	Edit	Text	Onti	one G	enerat	a Info	View	N 10/	indow	Help		-			-0.22	-0.23
14		-1.60	0.00	0.01	0.02	0.02	0.02	0.03	0 6	a riic	Luit	TEXL	Optil	JIIS G	enerau	- 1110	) viev	~ ~	muow	Tielp					-0.03	-0.03
15		-1.40	-0.02	-0.06	-0.09	-0.11	-0.14	-0.15	-0	1 1	À 📙		$ \langle \rangle$	NA	> 습			<u> </u>	7	-	az	1/		»	0.17	0.17
16		-1.20	-0.05	-0.12	-0.19	-0.24	-0.29	-0.33	-0 "					1.1			~			_	-				0.36	0.36
17		-1.00	-0.08	-0.18	-0.28	-0.36	-0.44	-0.49	-0							exam	ples.	xls							0.53	0.54
18		-0.80	-0.10	-0.23	-0.36	-0.47	-0.56	-0.63	-0							Z	Grid								0.69	0.70
19		-0.60	-0.12	-0.28	-0.43	-0.56	-0.67	-0.75	-0				3	191	\ <u>\</u>	115				2					0.81	0.82
20		-0.40	-0.13	-0.31	-0.47	-0.62	-0.74	-0.84	-0				2.5	110	عق	Zġ,	િંહ	0.0	0.75	Š.	0.9				0.91	0.92
21		-0.20	-0.14	-0.33	-0.51	-0.66	-0.79	-0.89	-0				2	0.7	$\geq$	$ \ge $					0.75				0.97	0.98
22		0.00	-0.14	-0.33	-0.52	-0.68	-0.81	-0.91	-0				1.5		0	0.10	<u> </u>			0-					0.99	1.00
23		0.20	-0.14	-0.33	-0.51	-0.66	-0.79	-0.89	-0				1	- 51		5	P	0.45	=0.3		0.45				0.97	0.98
24		0.40	-0.13	-0.31	-0.47	-0.62	-0.74	-0.84	-0				0.5	116	5	ૈલો	11//	15			0.3				0.91	0.92
25		0.60	-0.12	-0.28	-0.43	-0.56	-0.67	-0.75	-0				> 0	- 12	$\langle \cdot \rangle$	1)]]]		19	7)		0.15				0.81	0.82
26		0.80	-0.10	-0.23	-0.36	-0.47	-0.56	-0.63	-0				-0.5	1.6/1	60-	'///I	]]][[[	18	ונר		-0.15				0.69	0.70
27		1.00	-0.08	-0.18	-0.28	-0.36	-0.44	-0.49	-0					100		10	/////	$\sim$	16	J	-0.3				0.53	0.54
28		1.20	-0.05	-0.12	-0.19	-0.24	-0.29	-0.33	-0					0.15		-10-		$\leq$	0	s <b>/</b>   ·	-0.45				0.36	0.36
29		1.40	-0.02	-0.06	-0.09	-0.11	-0.14	-0.15	-0				-1.0	1	0.15	_0	+		-0.15	_	-0.75				0.1/	0.17
30		1.60	0.00	0.01	0.02	0.02	0.02	0.03	0				-2	100	3		10		0.	3	-0.9				-0.03	-0.03
31		1.80	0.03	0.08	0.12	0.15	0.18	0.21	0				-2.5	141	100	2		28	0.75						-0.22	-0.23
32		2.00	0.06	0.14	0.21	0.28	0.34	0.38	0				-3	101	1.0	γų	щи	47							-0.41	-0.42
33		2.20	0.08	0.20	0.30	0.40	0.48	0.54	0					-3	-2	-1	0	1	2	3					-0.58	-0.59
34		2.40	0.10	0.25	0.38	0.50	0.60	0.67	9	_	_	_	_			_	x	_	_	-	_		_		-0.73	-0.74
35		2.60	0.12	0.29	0.44	0.58	0.69	0.78	0.00	0.00	0.04	0.00	0.72	0.01	0.40	0.33	0.17	0.00	-0.17	-0.33	-0.40	-0.01	-0.72	-0.00	-0.84	-0.86
36		2.80	0.13	0.32	0.49	0.64	0.76	0.86	0.92	0.94	0.93	0.88	0.79	0.68	0.53	0.37	0.19	0.00	-0.19	-0.37	-0.53	-0.68	-0.79	-0.88	-0.93	-0.94
37		3.00	0.14	0.33	0.51	0.67	0.80	0.90	0.96	0.99	0.98	0.92	0.83	0.71	0.56	0.39	0.20	0.00	-0.20	-0.39	-0.56	-0.71	-0.83	-0.92	-0.98	-0.99

### 8.1.10 OneD Add-In command

Select one or more columns of amplitudes and produce a box-and-whisker plot. If the first row contains non-numeric entries, it is used as the label for each data group.

#### Initial plot settings:

The initial plot will be a median-based (rather than mean-based) box-and-whisker plot, with groups oriented vertically. To change the ranges represented by the box and/or whiskers or to edit the definitions of outliers and extremes, right-click on the plot and select "Format". To change the orientation of the plot, or the colors, line styles, and symbol styles used by various objects, select "Attributes" on that same "Box-and-Whisker Options" dialog.

To switch from a box-and-whisker plot to a dot graph, right-click on the plot and select "Dot Graph".

#### Example:

	A	В	С	D	E	F	G	Н		J	K	L	M	N	
1	OneD test														
2															
3	Select one or more columns, then select OneD on the DP						ot menu								
4	means=	2	3	4	5		6	7							
5	variance=	1	2	3	4		5	6							
6															
7		Group A	В	С	D	E	F								
8		1.837438	2.424028	2.907654	5.776275	7.9610	46 6.14032	4							
9		2.447373	3.76812	4.640204	4.387569	3.6681	88 4.1248	8							
10		2.010672	2.080101	2.958039	5.580847	3.7794	91 8.70039	8							
11		2.315142	2.571561	2.845142	3.001781	7.6288	53 7.55076	5							
12		2.114938	2.700185	3.620744	3.334292	6.8816	53 7.00812	9							
13		1.714119	3.675228	3.5212	4.556851	5.3726	57 7.65753	7							-
14		1.766427	3.167269	4.437085	3.250877	6.5 🔀	DPlot - [exa	mples (versio	n 1).xls]						
15		1.728425	3.630886	4.688568	5.646846	5.4	- โป้ []. [].	Taut Out	in an Minut	Mindau	Hala				a 📃
16		1.881939	2.942682	3.683448	5.353946	5.8		Text Opt	ions view	window	нер	- 1		^	
17		1.970081	3.421422	4.38862	6.991942	5.3	📄 🔁 🕻	- 🚖 🔍		à 🖹 I	🗟 📝 😡	2 🔶 🕻	7 1/	×	
18		2.374627	3.983027	4.840592	3.80178	8.2					<b>_</b>				
19		2.023952	2.281581	4.233249	3.172633	5.1			ex	amples (v	ersion 1).xl	S			
20		2.205995	3.668499	4.813042	6.833582	3.7				Un	eD				
21		2.051494	2.754911	3.751112	5.249124	5.8	9.6							<b>T</b>	
22		1.961342	3.833946	4.409809	4.677422	5.4	8.8	linimum							
23		1.931501	2.025096	3.561189	3.692732	5.4	8 7	5%				— T			
24		1.50239	2.537083	2.86191	6.955758	5	72 2	5%						_	
25		1.718867	2.348114	3.414096	4.761313	4.2		ledian			T				
26		1.919593	2.199569	5.437796	6.016404	5.3	0.4 0	utliers							
27		1.717596	2.964442	4.465522	4.617514	8.4	5.6	Audities		Т					
28		1.902197	3.498385	4.442349		7.5	4.8				-				
29		1.944453	2.304939	2.683707		4.0	4		-						
30		2.421358	2.272657	4.546576		8.3	3.2								
31		1.538787	2.48111	4.920818		5.5									
32		1.937596	3.052775	4.913968		5.1	2.4	-	L						
33		1.720518	2.280527	3.197636		8.1	1.6								
34		2.332568	3.976574	4.485219		6.7	0.8								
35		2.410728	3.535152	3.207764		6.2	Group	A I	в	С	D	E		F	
36		2.000092	2.777881	3.030032		5.0				)	(				
37		1.571572	3.304178	5.328311		4.1750	41 9.45562	2					-		
38		1.612702	2.598551	5.32142		5.7215	94 4.86907	9							

## 8.2 Transferring DPlot plots to other applications

#### Copy a metafile and/or bitmap to the Clipboard.

Select the <u>Copy Picture</u> command (or press CTRL+C) to copy a metafile and/or a bitmap image to the Clipboard. In the target application, select the Paste command (generally on the Edit menu) or Paste Special to choose among the available image formats.

A metafile preserves all of the information content of a plot and is much easier to edit than a bitmap. A metafile is device-independent, so the resolution of the image will be that of the final destination output device. DPlot produces both a standard metafile and an enhanced metafile. If the target application supports it, an enhanced metafile is generally preferable. However, DPlot will **not** produce either type of metafile for 3D views of 3D or 4D plots or for 2D views of surface plots that use shaded color bands. For those plot types a bitmap will be the only available format.

For problems with importing metafiles into another application, particularly into Microsoft Office products, please see the <u>Known Problems</u> section.

Note that a much more limited number of applications can support metafiles than bitmaps, and in some cases the metafile will be automatically converted to a bitmap.

The Copy command also creates a bitmap image and transfers that image to the Clipboard. The bitmap image created by the Copy command uses the dimensions specified by the <u>Copy Dimensions</u> command. Bitmaps are supported by a wide variety of applications. Bitmap images do not generally scale well relative to metafiles.

#### Save an image to a disk file.

Many graphic applications and word processors will allow you to import a picture file from disk. In Microsoft Word, for example, you can insert an image in a document by selecting Insert>Picture>From file. DPlot can export:

File	Description
WMF EMF	Standard Windows Metafile Enhanced Metafile
	The same restrictions mentioned above apply to WMF and EMF files: DPlot will not produce metafiles for 3D views of 3D or 4D data or for 2D views of surface plots that use shaded color bands.
BMP	Windows Bitmap Choose from 8- or 24-bit color depth, and optionally choose run-length encoding for 8-bit files. Though RLE-encoding of bitmaps is not often used, it is supported by most popular
GIF	graphics applications and results in a significantly smaller file for typical DPlot graphs. Graphics Interchange Format
	The image will be limited to 256 colors. For XY plots this is generally not a limitation, but may result in loss of detail in 3D images with many colors.
JPG	JPEG Picture For quality settings other than "Lossless", the resulting image will have artifacts typical of JPEG files. For a better quality plot in a usually smaller file. PNG is recommended
PNG	Portable Network Graphics
	Unlike JPEG, PNG compression is lossless. And unlike GIF, 24- or 32-bit images can be saved. Compression of DPlot plots is generally superior to the poorest quality JPEG files and is comparable to GIF. However, PNG is not as widely supported as any of the other bitmap formats.
TIF	Tagged Image File Format DPlot will always use LZW compression. Files saved with this plugin will generally be larger than the corresponding PNG image.
To save any c	of these formats, select the Save As command on the File menu, then select the desired

To save any of these formats, select the <u>Save As</u> command on the File menu, then select the desired format in the **Save As Type** box. For the bitmap formats you will then be prompted for a width and height (in pixels) and resolution of the image.

If the width or height values are set to 0 **and** the resolution is set to 100 or less dpi, the saved image will essentially be a screenshot. If the resolution is set to greater than 100 then font sizes, line widths, the plot size and other dimensions as shown on your display will be scaled up by *resolution/(screen resolution)*, so that the saved image will have the same proportions as the graph shown on your display. For best results with a resolution setting greater than 100, always specify the plot size with a <u>Size</u> command or the <u>Extents/Intervals/Size</u> menu command. Otherwise you risk generating an *Insufficient memory* error for high resolution settings.

If you are saving to an 8-bit color depth **and** your image has a colored background, you will generally get superior results if you check the *Improved color quantization* box. For large images (large in this case defined as wider than 3500 pixels), the *improved* quantization method will be used regardless of your selection here, primarily because the default method performs poorly with large images.

**NOTE:** For images with many unique colors and/or <u>antialiased lines and symbols</u> and for 3D views of 3D/4D data in which one of the light source models is used, you will get better results saving the image as 24- rather than 8-bits. If file size is paramount you can then use a dedicated image editor like <u>Adobe</u> <u>Photoshop</u> or <u>Paint Shop Pro</u> to reduce the color depth, with better results than you will get by saving to an 8-bit image in DPlot.

## 8.3 Transferring DPlot data to other applications

There are two methods for transferring DPlot data to other applications.

#### Copy data to the Clipboard.

You can copy the data values of a single curve to the Clipboard, then paste the data in the target application. See <u>Copy Data Values</u> command.

#### Use target application's DDE features.

If the target application can serve as a Dynamic Data Exchange (DDE) client, you can instruct DPlot to provide application information and/or data for one or more curves.

#### **System Topics**

The following Microsoft Excel formulas may be used to request information from DPlot:

<u>Formula</u>	Returns
=DPlot System!Topics	XYPlot
	System
	Plot title(s), if any
=DPlot System!SysItems	Topics
Or	SysItems
=DPlot!System!TopicItemList	Formats
	Status
	TopicItemList
	Help
=DPlot System!Formats	TEXT
=DPlot System!Status	Ready or Busy
=DPlot System!Help	See DPlot Help "Transferring DPlot data
	to other applications"
=DPlot System!DecimalSymbol	Decimal symbol character used by DPlot on
	input; typically a period or a comma.
=DPlot System!DocumentNumber	Index of the document associated with this
	DDE conversation, 1-32
=DPlot System!	Number of open document windows.
NumberOfDocuments	
=DPlot System!LastNewCaption	Returns the most recently set document
	caption (generally the caption for the most
	recently created document)
=DPlot System!LastNewDocNum	Similar to LastNewCaption; returns the
	document index (1-32) of the most recently
	created document.
=DPlot System!Version	DPlot version number
=DPlot System!LastCurveFit	Returns the polynomial curve fit last
	generated by using the <u>Polynomial Curve Fit</u>
	menu command or by a <u>Curverit</u> macro/DDE
	command.

**NOTE:** For formulas that return more than one value, with values separated either by tabs (e.g. TopicItemList) or carriage returns (e.g. LastCurveFit), you must first select the desired cells in the worksheet, enter the formula, then press <Ctrl>+<Shift>+<Enter>.

The syntax in Visual Basic programs for the above commands is:

```
<control>.LinkTopic="DPlot|System"
<control>.LinkMode=2
<control>.LinkItem=<item to retrieve, e.g. "NumberOfDocuments">
<control>.LinkRequest
<control>.LinkMode=0
```

where <*control*> is a text box or label.

#### **Document Topics**

If one or more plots are currently open, you can use the target application's DDE features to request information about those plots. For example, if you open the DPlot file c:\my data\squares.grf, then the following request items can be used to request information about that plot:

Request ItemReturnsCurve00111

	2 4
	3 9
	4 16
FileName	c:\my data\squares.qrf
FileTitle	squares.qrf (text appearing in the plot title bar)
fname	squares (filename minus the extension)
Formats	TEXT
Legend(n)	Legend title for $n=0$ or legend entry for curve $n$ ( $n=1$
0 ()	through number of curves)
ListPeaks	Curve 1
	Minimum = 1 at X = 1
	Maximum = 16 at $X = 4$
	Mean = 7.5
	Standard deviation = 6.557439
	Standard error = 0
MaxCurves	maximum number of curves
MaxPoints	maximum number of points/curve
NumberOfCurves	1
NumberOfPoints	4
Path	c:\my data\ (folder location of the saved plot, including a
	trailing backslash)
Title <i>n</i> ( <i>n</i> =1-4)	Title lines
TopicItemList	Formats
·	MaxCurves
	MaxPoints
	NumberOfCurves
	NumberOfPoints
	TopicItemList
	Curve01
XAxisLabel	X axis label
YAxisLabel	Y axis label
ZAxisLabel	Z axis label

To retrieve this information in Excel, use a formula of the form "=application|topic!item". The "application" in all cases is "DPlot". The "topic" is the file title of a plot (the text appearing in the document title bar. To retrieve the X axis label from squares.grf, for example, use "=DPlot|squares.grf! XAxisLabel".

If interacting with DPlot via the <u>DPLOTLIB DLL</u>, all of the information shown above can be retrieved with a call to DPlot_Request. For example, to retrieve the character used as a decimal symbol in C:

```
char sDecimal[2];
int dwSize=sizeof(sDecimal);
DPlot Request(0,"DecimalSymbol",sDecimal,&dwSize);
```

The plot title is the text that appears in the caption of the plot window. To retrieve the caption of the last document created, you can use the LastNewCaption request mentioned above. For example, if your application uses DPlot to produce Histograms, the first plot produced will have a caption of 'Histogram'. The 2nd histogram (assuming the 1st one is still open) will have a caption of 'Histogram1', the 3rd a caption of 'Histogram2', etc. However, there is no good way for you to determine programmatically whether an end user might have closed a document that your application produces, so it is all but impossible to predict what the caption of a new document might be. If you use LastNewCaption immediately after taking some action that produces a new document, you can be assured that the return value is the caption you are interested in. You can also retrieve the document index (a number, 1-32) of the last document created by requesting LastNewDocNum, which is more useful for programs using DPLOTLIB's DPlot_Command and other functions. To request the above information knowing the document index but not the document's caption, you can use the form **Doc**<*n*> for the DDE topic, where *n* is the document index (1-32).

Data values submitted by DPlot are separated by <TAB> characters. A <CR><LF> sequence separates X,Y pairs and the LastCurveFit output.

#### See also

Request macro function

501

# Index

## - 3 -

3D intersections2893D scale factors, automatic2633D scatter plot2813D scatter plots4933D surface plots492, 4953D view323

## - A -

AASHTO 209, 360 AAxisLabel macro command 315 About macro command 315 abs 441 absolute value 441 441 acos acosh 442 Activate macro command 315 add a constant to 171 Х Y 171 Ζ 172 add surfaces 291 AddCurves macro command 316 Add-In 480 Adobe PDF 72 and 443 animation 302 anorthosite 202 antialiasing 264 appearance 190 append data 142 AppHide macro command 316 apply a transform to Х 171 Y 171 Ζ 172 AppMaximize macro command 316 AppMinimize macro command 316 AppRestore macro command 316 AppShow macro command 316 arccosine 441 arcsine 443

arctangent 445 area under a curve 84, 185, 272 arrange 304 array size 153 ASCII file 142 443 asin 444 asinh 228 asymptotes atan 445 445 atan2 445 atanh ATI Mach 32 display adapter 472 automatic 3D scale factors 263 automatic scaling 316 317 automatic sizing 316 AutoScale macro command AutoSize macro command 317 average 287, 317 average curves 268 293 average surface AverageAll macro command 317 AverageAndReplace macro command 317 Axes At Zero macro command 317 axis direction 88 axis label orientation 88 axis labels 92 AxisArrowheads macro command 317 AxisColor macro command 317 AxisType macro command 318 azimuth 244, 332

## - B -

B/W printers 86 background color 319 background image 260, 384 56, 98, 192, 284, 318, 364 bar chart bar chart fill pattern 356 bar chart options 205 bar graph 98 BarChartStyle macro command 318 BarWidth macro command 318 base-10 logarithm 454 baseline 167, 296 baseline shift 318 BaselineShift macro command 318 batch operations 73, 86 **Bessel function** first kind 446

**Bessel function** second kind 447 besseli 446 bessly 447 bitmap 142, 146, 388, 496 443, 459, 460, 470 bitwise operations BkColor macro command 319 BkPlotColor macro command 319 BMP 142, 146, 388, 496 box and whisker plot 256 495 box-and-whisker plots bubble plot 232, 320 Bubbles macro command 320 bug report 476

## - C -

calendar date 91 call macro 320 Call macro command 320 Candlestick chart 235 Caption macro command 320 cascade 304 cdf 448 449 ceil ceiling 449 circle 75 ClearEditFlag macro command 321 clinopyroxene 202 clip art 262 clipboard 157, 158, 159, 160, 334, 376, 496, 497 ClipY macro command 321 close 153 close document 147, 346 149, 190, 227 color Color macro command 321 321 ColorBasis macro command ColorCycle macro command 321 ColumnsAre macro command 322 CombineCurves macro command 323 command line 108 comma-separated values 69 common logarithm 454 18, 31, 36, 39, 45, 46, 49, 50, common questions 52, 54, 56, 58, 60, 62, 64, 68, 69, 71, 72, 73, 74, 86, 88, 89, 90, 91, 92, 107, 111, 112 Compare macro command 323 compare records 267 concave edges 64

concave surface plots 258 concave surfaces 337 contour interval 327 contour plot 243, 244, 253, 258, 279 Contour3D macro command 323 ContourAxes macro command 323 ContourBarWidth macro command 323 ContourColorScheme macro command 324 ContourCustomColors macro command 325 ContourCustomLevels macro command 325 ContourDrawSides macro command 325 ContourGrid macro command 326 ContourLabelLines macro command 326 ContourLegend macro command 327 327 ContourLegendFrequency macro command ContourLevels macro command 327 ContourLighting macro command 328 328 ContourMethod macro command ContourScales macro command 331 ContourView macro command 332 333 ContourViewChange macro command copy 157, 158, 334, 496, 497 CopyPicture macro command 334 CopyText macro command 334 449 cos cosh 450 crop images 386 CSV file 69 CSV files 18, 123 cumulative distribution function 192, 448 224 currency 268, 334 curve fit curve fitting 334 CurveFit macro command 334 CurveLabel macro command 334

## - D -

d/m/y date format 265 data generating new 267 data entry 68 data processing 167, 231, 318, 334, 335 DataProcess macro command 334 DataProcessLabels macro command 335 date 451 date format 335 date format on input 265 451, 455, 469, 470 date functions

Index

503

date separator 266 DateFormat macro command 335 dates 224 day 451 DDE 160, 161, 497 DDEML 307 decimal symbol 266, 385 DecimalDigits macro command 335 decimate data 163 default conditions 123 DeferMinMaxCheck 336 258 define boundary of surface plots Deletelmage macro command 336 DeleteLabel macro command 335 336 DeleteNote macro command DeleteRefLine macro command 336 337 DeleteThinTriangles macro command DenseGridLines macro command 337 dependent scales 49 derivative 273 different error values 95 273 differentiate 337 Differentiate macro command Directory macro command 337 265 disable popup coordinates display data values 298 distance between points 296 distribution 112 divide surfaces 291 DivideCurves macro command 338 DocHide macro command 338 DocMaximize macro command 338 338 DocMinimize macro command DocRestore macro command 339 339 DocShow macro command dot graph 257 dot graphs 495 **DPlot file** 123, 132, 142 DPLOTLIB 480 drag and drop 73 202 dunite

## - E -

edit 162 edit data values 162, 172 Edit menu Baseline Shift 167 Copy Data Values 158

Copy Dimensions 158 Copy Picture 157 Decimate 163 162 Edit Data Edit Mode 172 Equal Intervals 168 Filter Data 164 Insert Link 161 List Editions 167 Operate on X 171 Operate on Y 171 Operate on Z 172 Paste 159 Paste Link 160 **Remove Random Spikes** 173 Smooth 167 Swap X,Y 174 339 EditAddToX macro command EditAddToY macro command 339 EditDecimate macro command 339 339 EditEqualInts macro command EditErase macro command 339 EditEraseData macro command 340 EditFilter macro command 340 EditMultiplyX macro command 341 EditMultiplyY macro command 341 EditNormalize macro command 341 EditOperateX macro command 341 EditOperateY macro command 341 EditOperateZ macro command 341 342 EditRemoveSpikes macro command EditRemoveTrend macro command 342 EditReplaceData macro command 342 EditSmooth macro command 343 EditSort macro command 343 EditSubtractMean macro command 343 EditSwapXY macro command 343 EditTruncate macro command 343 elevation 244, 332 ellipse 75 344 Enable macro command enable/disable DPlot 344 Encapsulated PosteScript 71 end user license agreement 115 environment variables 347 EPS 71 erase curve 339, 340 erf 451 erfc 452

344

error bars 95, 240, 287, 344 471 error messages ErrorBar macro command 344 ErrorBarPenWidth macro command ErrorBarWidth macro command 344 ErrorMessages macro command 345 errors 472 EULA 115 evaluation license 114 example plot EX01 46 49 example plot EX02 example plot EX03 50 example plot EX04 52 example plot EX05 54 example plot EX08 56 58 example plot EX09 example plot EX10 60 example plot EX14 62 example plot EX16 64 example plots 45 Excel 480 153 exit exit DPlot 347 exiting DPlot 74 452 exp Explorer open files from 73 exponent 452 export 496 export data 158 214 extents specifying plot 369

## - F -

fact 453 factorial 453 Fast Fourier Transform 274 274, 345 FFT FFT macro command 345 FFTMethod macro command 345 FFTPhase macro command 345 134, 135, 136, 138, 139, 140 file formats file import 140, 422 File menu 122 1,2,3,4 153 Append 142 147 Close Exit 153

Get Preferences 147 Macro 150 123 New Open 123 Print 149 **Resize Arrays** 153 Save 142 Save As 142 Save Preferences 148 file type 354 specifying 345 FileAppend macro command FileArrays macro command 346 FileClose macro command 346 347 FileExit macro command FileMacro macro command 347 FileNew macro command 347 347 FileNewEx macro command FileOpen macro command 347 FilePageSetup macro command 349 FilePrint macro command 350 FilePrintMultiple macro command 351 FileSaveAs macro command 352 FileType macro command 354 fill areas 355 fill between curves 220 FillBetween macro command 355 FillStyle macro command 356 filter digital 340 164, 166, 173, 342 filters FindIntersections macro command 356 453 floor font size 356 FontPoints macro command 356 182, 356, 407 fonts 356 ForAllDocuments macro command ForFilesIn macro command 357 fpower 453 fractions 224 frequency domain 274 function 275, 278, 439 FunctionOfXY1Y2 macro command 358 functions 439 358 FXY1Y2 XFrom macro command

## - G -

gabbro 202
505

GCD 453 GeneralOptions macro command 358 generate data 275, 278, 279, 281 Generate menu 267 Average All Curves 268 Average Curve with Error Bars 287 Average Surfaces 293 Differentiate a Curve 273 FFT 274 Find Best-Fit Plane 291 Histogram 284 Integrate a Curve 272 Median of All Curves 268 Polynomial Curve Fit 268 Subtract a Plane 291 Switch Independent Variable 293 X=f(T), Y=g(T)278 X=f(T), Y=g(T), Z=h(T)281 XY Plot from 3D Surface 289 Y=f(X)275 Y=f(X,Y1,Y2,...)282 Y = Poly(Y)282 Z=f(X,Y)279 Z3=f(Z1,Z2)291 359 GenerateMesh macro command GetPreferences macro command 360 getting started 17, 18, 31, 36, 39 GIF 142, 146, 388, 496 Gouraud 328 grain size distribution plots 102, 360, 362 209 grain size option GrainSizeFlags macro command 360 GrainSizeFlagsEx macro command 362 GrainSizeOther macro command 362 graph a circle 75 graph an ellipse 75 graph an equation 74 graph paper 86 graphing an equation 46 graphing an inequality 76,80 grid line color 321 grid line width 91 arid lines 211, 326, 337, 362, 363 GridLines macro command 362 GridLinesOnTop macro command 363 GridLineType macro command 363

## - H -

harzburgite 202 help 112 363 HelpContents macro command HelpSearch macro command 363 hide 338 hide a curve 111, 363, 432 hide curves 266 hide window 316 HideCurve macro command 363 hiding the legend 367 hiding the menu 369 histogram 284, 318, 364 Histogram macro command 364 holes in surface plots 258 HP LaserJet 4 472 hyperbolic cosine 450 hyperbolic functions 442, 444, 445, 450, 465, 468 hyperbolic sine 465 hyperbolic tangent 468

#### - | -

image export 146, 388 images 336, 365 importing data 69, 159, 160, 161 importing from DOS 69 independent Y scales 52 IndependentTicks macro command 364 inequality, graphing 76.80 Info menu 296 297 Interpolate 299 List Peaks Scan X,Y 298 InfoInterpolate macro command 364 InfoScan macro command 365 input date format 265 insert image 262 inserted images 336, 365 InsertImage macro command 365 inserting data 365 InsertLink macro command 365 InsertXY macro command 365 integrate 231, 272 366 Integrate macro command interpolate 168, 297, 364

intersections finding 356 invert axis 88 InvertAxis macro command 366 IUGS mafic 202 IUGS ultramafic 202

#### - J -

JPEG 142, 388, 496 JPG 142, 146, 388, 496

# - K -

keyboard 430 kurtosis 299

# - L -

label 335, 366 label points 172, 180 LabelDirection macro command 366 labeling axes 419 326 labeling contours labeling points 411, 412, 413 labeling second Y axis 421 labeling X axis 418 labeling Y axis 421 labeling Z axis 422 labels 177, 180, 182, 334 LCM 454 least squares 291, 334 least squares fit 268 legend 177, 327, 366, 367 Legend macro command 366 LegendParams macro command 366 LegendShow macro command 367 Iherzolite 202 Licensing 114 lighting 328 line color 227 line size 212 line style 212 line types 367 line width 91, 368 LineType macro command 367 LineWidth macro command 368 link 379

link data from DDE server 365 list peak values 299 ListPeaksOptions macro command 368 454 log log scales 217 log10 454 logarithm 454 logarithmic scale 368, 394 logarithmic scales 192, 217 logo images 262 LogTicks macro command 368 Lump macro command 368

## - M -

macro commands AAxisLabel 315 315 About Activate 315 AddCurves 316 AppHide 316 AppMaximize 316 AppMinimize 316 AppRestore 316 AppShow 316 AutoScale 316 AutoSize 317 AverageAll 317 AverageAndReplace 317 Axes At Zero 317 AxisArrowheads 317 AxisColor 317 AxisType 318 BarChartStyle 318 BarWidth 318 BaselineShift 318 BkColor 319 BkPlotColor 319 **Bubbles** 320 320 Call Caption 320 ClearEditFlag 321 ClipY 321 Color 321 ColorBasis 321 ColorCycle 321 ColumnsAre 322 323 CombineCurves Compare 323

© 2014 HydeSoft Computing, LLC

507

macro commands Contour3D 323 ContourAxes 323 ContourBarWidth 323 ContourColorScheme 324 ContourCustomColors 325 ContourCustomLevels 325 ContourDrawSides 325 ContourGrid 326 326 **ContourLabelLines** ContourLegend 327 ContourLegendFrequency 327 ContourLevels 327 ContourLighting 328 **ContourMethod** 328 ContourScales 331 **ContourView** 332 **ContourViewChange** 333 CopyPicture 334 CopyText 334 CurveFit 334 CurveLabel 334 DataProcess 334 335 DataProcessLabels DateFormat 335 DecimalDigits 335 **DeferMinMaxCheck** 336 Deletel mage 336 335 DeleteLabel DeleteNote 336 DeleteRefLine 336 **DeleteThinTriangles** 337 DenseGridLines 337 Differentiate 337 Directory 337 **DivideCurves** 338 338 DocHide DocMaximize 338 338 DocMinimize DocRestore 339 DocShow 339 EditAddToX 339 EditAddToY 339 EditDecimate 339 EditEqualInts 339 EditErase 339 EditEraseData 340 EditFilter 340 EditMultiplyX 341

EditMultiplyY 341 EditNormalize 341 EditOperateX 341 EditOperateY 341 EditOperateZ 341 EditRemoveSpikes 342 342 EditRemoveTrend EditReplaceData 342 EditSmooth 343 EditSort 343 EditSubtractMean 343 EditSwapXY 343 EditTruncate 343 Enable 344 ErrorBar 344 ErrorBarPenWidth 344 ErrorBarWidth 344 ErrorMessages 345 FFT 345 FFTMethod 345 FFTPhase 345 FileAppend 345 FileArrays 346 FileClose 346 FileExit 347 FileMacro 347 347 FileNew FileNewEx 347 FileOpen 347 FilePageSetup 349 FilePrint 350 **FilePrintMultiple** 351 FileSaveAs 352 FileType 354 FillBetween 355 FillStyle 356 **FindIntersections** 356 **FontPoints** 356 ForAllDocuments 356 ForFilesIn 357 FunctionOfXY1Y2 358 FXY1Y2_XFrom 358 GeneralOptions 358 GenerateMesh 359 GetPreferences 360 GrainSizeFlags 360 GrainSizeFlagsEx 362 GrainSizeOther 362 GridLines 362

macro commands GridLinesOnTop 363 GridLineType 363 **HelpContents** 363 363 HelpSearch HideCurve 363 364 Histogram IndependentTicks 364 InfoInterpolate 364 InfoScan 365 InsertImage 365 InsertLink 365 InsertXY 365 366 Integrate InvertAxis 366 LabelDirection 366 Legend 366 LegendParams 366 LegendShow 367 LineType 367 LineWidth 368 **ListPeaksOptions** 368 368 LogTicks Lump 368 ManualScale 369 369 Medians 369 MenuHide MenuShow 370 MenuUpdates 370 MoveCurves 370 MultipleXAxes 370 **MultipleYAxes** 371 **MultiplyCurves** 372 NextDocument 372 NextFile 372 NoteButtonText 372 NoteOptions 373 NotePosition 373 373 NoteTabstops NoteText 373 NumberFormat 374 NumTicks 375 OHLC 375 **OverlayDocument** 375 Paste 376 PasteLink 376 376 Pause **PolarPlotStyle** 376 PostponeRedraw 377

Preserve3DScaleFactor 377 **ProbabillityFlags** 378 378 RefLine RemoveLink 379 ReorderCurves 379 Repaint 379 RunPlugin 380 **SavePreferences** 380 SaveUndo 381 Scale 381 ScatterPlotLines 382 383 SelectCurve SetBitmapDims 385 385 SetClientRect SetDecimalSymbol 385 SetFilename 386 SetImageCrop 386 SetMetafileDims 386 386 SetOutputWindow SetPanFraction 388 SetPluginImageDims 388 SetWindowPos 390 Shell 392 ShowCaption 392 393 ShowCurve 393 ShowDate 393 Size SplitDateTime 394 SquareLogCycles 394 StartStop 394 395 Stat_AddData Stat AddGroup 395 Stat Box 395 Stat_BWOptions 395 Stat DotOptions 396 Stat_DrawLegend 396 Stat_GrandMean 396 Stat_GroupInfo 397 Stat LimitsAsCL 397 Stat MeanFormat 397 Stat_Outliers 397 Stat_PlotType 398 Stat Refline 398 398 Stat_Rotated Stat Scaling 398 Stat_Whiskers 398 SubtractCurves 399 SwitchX 399 SymbolCycleColors 402

macro commands SymbolFillBackground 402 SymbolFrequency 402 SymbolOpacity 402 SymbolOutline 404 SymbolOutlineColor 404 SymbolSize 405 SymbolSizeUnits 405 SymbolType 405 TextFont 407 **TextNote** 407 TextNote3D 408 TextNoteEx 409 TextNoteLeader 411 TextPointLabel 411 TextPointLabel3D 412 TextPointLabelLeader 413 413 TickInterval **TickMarks** 414 TimeFormat 414 Title1 414 Title2 414 414 Title3 Title4 415 TriangleFlags 415 TripartiteFlags 415 UseNameAsLegend 415 VectorPlot 416 **VectorPlotColors** 416 **VectorPlotOptions** 416 ViewRedraw 417 ViewStatusBar 417 ViewToolBar 417 WindowCascade 417 WindowTile 418 XAxisLabel 418 XLabel 418 XY 418 XYLabel 419 XYXY 419 XYZ 420 XYZ_To_XY 420 XYZEx 420 XYZRegen 421 Y2AxisLabel 421 Y2TopDown 421 YAxisHorizontal 421 421 YAxisLabel YCutoff 421

ZAxisLabel 422 macro editor 150 macro loops 356, 357, 372 mafic rock 202 ManualScale macro command 369 349 margins max 454 maxima 185, 299 maximize 316, 338 maximize at startup 263 454 maximum 153 maximum number of curves maximum number of points 153 maximums 185, 299 185, 299 mean median 268 Medians macro command 369 menu commands 122 MenuHide macro command 369 370 MenuShow macro command MenuUpdates macro command 370 142, 386, 388, 496 metafile middle mouse button 266 min 454 185, 299 minima minimize 316, 338 454 minimum minimums 185, 299 minor grid lines 337 mod 455 modify X 171 modify Y 171 modify Z 172 month 455 432 mouse MoveCurves macro command 370 multiple axes 217, 370, 371 MultipleXAxes macro command 370 MultipleYAxes macro command 371 multiply surfaces 291 multiply X by a constant 171 multiply Y by a constant 171 multiply Z by a constant 172 MultiplyCurves macro command 372

natural logarithm 454 NextDocument macro command 372 NextFile macro command 372 noise removal 164, 166 non-convex 3D meshes 91 normal distribution 456, 457 normalize 341 456 normdist normdistcdf 456 norminv 457 457 normrand 459 not 336 notations 182 note 372 NoteButtonText macro command NoteOptions macro command 373 373 NotePosition macro command NoteTabstops macro command 373 NoteText macro command 373 number format 91, 224, 374 number of points 296 number orientation 88 NumberFormat macro command 374 NumTicks macro command 375

### - 0 -

235 OHLC chart OHLC macro command 375 olivine 202 opacity 212, 402 open file 123, 132 Options menu 190, 256, 257 **Amplitude Limits** 228 **Bar Chart Options** 205 **Bubble Plot** 232 Colors 227 **Contour Options** 244 **Data Processing** 231 258 Define Boundary Error Bars 240 Extents/Intervals/Size 214 Fill Between Curves 220 General 263 Generate Mesh 243 Grain Size Options 209 Grid Lines or Box 211 Linear/Log Scaling 192 Log Scale Tick Labels 217 Multiple Axes 217 Number Format 224

OHLC chart 235 Polar Options 204 **Reference Lines** 241 Scatter Plot Options 253 Square Log Cycles 217 Symbol/Line Styles 212 Triangle Plot Options 202 **Tripartite Options** 210 Vector Plot 233 460 or order form 118 ordering info 116 orthopyroxene 202 OverlayDocument macro command 375 overlays 375

#### - P -

paint 355 pan and zoom 111 paste 159, 160, 376 376 Paste macro command PasteLink macro command 376 Pause macro command 376 PDF 72, 350 peak values 296 pen width 212 244, 248, 332 perspective phase angle 345 Phong 328 plagioclase 202 302 playback plot corners 214 214, 393 plot size plugin 422 plugins 140 PNG 142, 146, 388, 496 point labels 60 polar chart 97 polar chart options 204 polar coordinates 97, 192 polar plot 97, 376 50 polar plots PolarPlotStyle macro command 376 popup coordinates, disable 265 PostponeRedraw macro command 377 **PowerPoint** 472 preferences 92, 360, 380 Preserve3DScaleFactor macro command 377

511

print 149 print document 350, 351 print multiple plots on same page 86 printer problems 472 printing gray scale 149 probability plot 192 ProbabilityFlags macro command 378 programmer's reference 306 programming 150, 306, 307 pyroxene 202

## - R -

randbetween 463 random number 457, 463, 464 read data points 298 read file 123, 132 real-time graphing 386 reduce data 163 reference 430 reference lines 241, 336, 378 RefLine macro command 378 registered license 115 registration 120 RemoveLink macro command 379 ReorderCurves macro command 379 Repaint macro command 379 replace data 342 reporting errors 476 restore window 316, 339 reverse axis 88 reversing axes 366 right click 432 rnd 464 rotate a 3D plot 430 round 464 RunPlugin macro command 380

# - S -

save 142 save document 352 save file 142 SavePreferences macro command 380 SaveUndo macro command 381 scale 331 Scale macro command 381 scaling 190, 192

scan data 298 scanning data 365 scatter plot 92, 281 ScatterPlotLines macro command 382 scientific notation 91 383 SelectCurve macro command sequential time values 265 385 SetBitmapDims macro command SetClientRect macro command 385 385 SetDecimalSymbol macro command 386 SetFilename macro command SetImageCrop macro command 386 SetMetafileDims macro command 386 SetOutputWindow macro command 386 SetPanFraction macro command 388 SetPluginImageDims macro command 388 92 settings SetWindowPos macro command 390 shading 328 Shell macro command 392 167 shift shock spectra 54, 192, 210 shortcuts 430 show window 316, 339 ShowCaption macro command 392 ShowCurve macro command 393 ShowDate macro command 393 464 sign significant figures 335 sin 464 465 sinh Size macro command 393 skewness 299 slope 273 slope between points 296 smooth curve 89 smoothing 167, 343 sort data 343 spacing 168 constant sparse data 89 164, 173, 342 spike removal spin a 3D plot 430 SplitDateTime macro command 394 sart 466 square log cycles 217 square root 466 394 SquareLogCycles macro command standard deviation 185, 287, 299

standard error 287, 299 StartStop macro command 394 startup options 263 Stat AddData macro command 395 Stat AddGroup macro command 395 Stat Box macro command 395 Stat BWOptions macro command 395 396 Stat DotOptions macro command Stat DrawLegend macro command 396 396 Stat GrandMean macro command 397 Stat GroupInfo macro command 397 Stat LimitsAsCL macro command Stat MeanFormat macro command 397 Stat Outliers macro command 397 398 Stat PlotType macro command 398 Stat Refline macro command Stat Rotated macro command 398 398 Stat Scaling macro command Stat Whiskers macro command 398 300 status bar 291 subtract plane subtract surfaces 291 399 SubtractCurves macro command sum function 466 summation function 466 surface fit 291 surface plot 58, 258, 279, 289, 291, 293, 328, 337 surface plot with concave edges 64 swap axes 174 swap coordinates 343 switch axes 174 switch parameters 293, 399 SwitchX macro command 399 symbol 92, 212 symbol size 212, 405 symbol style 212 SymbolCycleColors macro command 402 SymbolFillBackground macro command 402 SymbolFrequency macro command 402 SymbolOpacity macro command 402 SymbolOutline macro command 404 SymbolOutlineColor macro command 404 symbols 402, 404, 405 SymbolSize macro command 405 SymbolSizeUnits macro command 405 SymbolType macro command 405

## - T -

467 tan tangent 467 tanh 468 Taylor series 466 ternary plot 62, 192 ternary plot options 202 Text menu Add/Edit Note 182 Fonts 182 Label Points 180 Legend/Labels 177 Title/Axes 175 X,Y,Z Labels 188 text note 407, 409 336, 372, 373 text notes text placeholders 185 text shortcuts 185 TextFont macro command 407 TextNote macro command 407 TextNote3D macro command 408 TextNoteEx macro command 409 TextNoteLeader macro command 411 TextPointLabel macro command 411 TextPointLabel3D macro command 412 TextPointLabelLeader macro command 413 tick mark interval 107, 375, 413 tick mark placement 414 tick mark size 414 tick marks 214 tick marks, programming 364 TickInterval macro command 413 TickMarks macro command 414 142, 146, 388, 496 TIF TIFF 142, 388, 496 tile 304 tiling windows 417, 418 time format 414 time, sequential 265 TimeFormat macro command 414 224 times Title1 macro command 414 Title2 macro command 414 Title3 macro command 414 Title4 macro command 415 303 toolbar tranferring data values to other applications 158 transferring graphics to other applications 157 transparency 146, 388 trend removal 342 triangle plot 62, 192 triangle plot options 202 triangle plot, symbols 264 TriangleFlags macro command 415 trigonometric functions 441, 443, 445, 449, 464, 467 tripartite grid 192 tripartite options 210.415 tripartite plots 54 TripartiteFlags macro command 415 troctolite 202 TrueType 149 truncate data 343

## - U -

Udden-Wentworth 209, 360 ultramafic rock 202 undo 263 unformatted binary files 130, 131 unformatted file 142 unique error values 95 204, 209, 217, 370, 371 units updates to DPlot 476 **USDA Soil Classification** 202 UseNameAsLegend macro command 415

### - V -

vector plot 233. 416 VectorPlot macro command 416 VectorPlotColors macro command 416 VectorPlotOptions macro command 416 view angles 332 View menu 300 302 Playback ViewRedraw macro command 417 ViewStatusBar macro command 417 ViewToolBar macro command 417

### - W -

websterite 202 weekday 469 weeknum 469 wehrlite 202 Welch 163 window menu 304 WindowCascade macro command 417 WindowTile macro command 418 WinExec 306

# - X -

x axis label 418 X axis, multiple units 370 186 X(i,j) placeholder X=f(T), Y=g(T), Z=h(T)281 XAxisLabel macro command 418 XLabel macro command 418 xor 470 **XTYP EXECUTE** 307 XY macro command 418 XY plots 486, 487, 491 XYLabel macro command 419 XYXY macro command 419 420 XYZ macro command 420 XYZ To XY macro command XYZEx macro command 420 XYZRegen macro command 421

## - Y -

Y axis 217 y axis label 421 Y axis, multiple units 371 Y(i,j) placeholder 186 Y2AxisLabel macro command 421 Y2TopDown macro command 421 YAxisHorizontal macro command 421 YAxisLabel macro command 421 YCutoff macro command 421 470 vear

# - Z -

z axis label 422 Z=f(X,Y) 279 ZAxisLabel macro command 422 zoom 300 zoom and pan 111