XMGR5 Users Manual

K. R. Jones[1]
J. E. Fisher

March 1997

Idaho National Engineering and Environmental Laboratory
Nuclear Systems Analysis Technologies Department
Lockheed Martin Idaho Technologies Company
Idaho Falls, Idaho 83415

[1]Currently at Scientech, Inc., Rockville, MD

DISCLAIMER

Portions of this document may be illegible electronic image products. Images are produced from the best available original document.

PREFACE

The XMGR5 software package is derived from an XY plotting tool for unix workstations called ACE/gr. The interactive version of ACE/gr is xmgr and includes a graphical interface to the X Windows system. Enhancements to the xmgr plotting tool have been developed which import, manipulate, and plot the output data from the RELAP5/MOD3 light water reactor thermal-hydraulic systems code and also from the MELCOR, FRAPCON, and SINDA codes, and from U.S. Nuclear Regulatory Commission (NRC) databank files. The resulting product is called XMGR5.

Sections 1 through 4 of the XMGR5 Users Manual contain the instructions for operating xmgr and are taken from a document entitled:

> Paul J. Turner, *ACE/gr Users Manual, Graphics for exploratory data analysis*
> Center for Coastal and Land-Margin Research
> Oregon Graduate Institute of Science and Technology
> 19600 NW von Neumann Dr., Beaverton, Oregon, 97006-1999

The following version information and copyright notices appear in the ACE/GR documentation:

Section 5 of the XMGR5 Users manual contains instruction in the use of extensions made to the ACE/gr software. These extensions include interfaces which directly import data channels from RELAP5 and other computer codes and from the NRC databank files, additions to the xmgr command interpreter, an equation interpreter which performs arithmetic and boolean operations on data sets, and the capability for tabular lookup from the RELAP5 two phase fluid property tables.

CONTENTS

TABLES

XMGR5 Users Manual

1.0 INTRODUCTION

ACE/gr is an XY plotting tool for workstations or X-terminals using X. A few of its features are:

> User defined scaling, tick marks, labels, symbols, line styles, colors.
> Batch mode for unattended plotting.
> Read and write parameters used during a session.
> Polynomial regression, splines, running averages, DFT/FFT, cross/auto-correlation.
> Hardcopy support for PostScript, HP-GL, and FrameMaker .mif format.

While ACE/gr has a convenient point-and-click interface, most parameter settings and operations are available through a command line interface (found in Files/Commands).

1.1 Terminology

Sets - Sets are collections of points, with optional associated values at these points. The associated values can be used to display error bars, high-low open-close plots and several other features. Sets are connected to a graph, and, by default, there are 30 sets per graph and each set is numbered from 0 to the maximum number of sets less one. The number of points in a set is limited by the size of virtual memory and are numbered from 1 to N where N is the total number of points in the set. Operations to manipulate sets are found in **Data/Set operations**, and features such as line styles, symbol types, and others can be changed from **Graph/Symbols**.

Graphs - Graphs are the collection of data sets, tick marks, titles, etc. drawn to display the data. By default, there are 10 graphs with each graph owning 30 sets (both the number of graphs and the number of sets are adjustable via command line parameters). Operations on graphs are found in the **Graphs/Graph operations** popup.

There are numerous popups accessible through Graphs pulldown menu that allow the manipulation of graph parameters. Most graph operations work on what is referred to as the "current graph", marked by small filled rectangles at the corners of one graph. The 10 graphs available by default in this version of ACE/gr are numbered from 0 to 9.

Regions - Regions are sections of the graph defined by the interior or exterior of a polygon, or a half plane defined by a line. A region defined by a line can be above, below, to the right, or left of the line. Defining regions and operations on points inside regions are found in **Data/Region operations**.

Parameters - Parameters are the settings of symbols, line styles, colors, fonts, etc. used to define graphs and the display of the active sets.

2.0 USING ACE/gr

2.1 Execution

The interactive **ACE/gr** program is called **xmgr**, if used in batch mode, grbatch. If grbatch is not found, contact your system administrator, or see the section on installation. For grbatch to work properly it is important that it be called grbatch as it is the name that distinguishes xmgr from grbatch. They are the same program, but when executed as grbatch, the X interface is not initialized and grbatch terminates at the end of execution without user intervention.

Please note that grbatch does not read the any X resource files, so that it can be run from dumb terminals and over phone lines. Don't expect resource settings used for xmgr to work for grbatch. See the section on setting defaults using a .xmgrrc file.

2.2 Command Line Parameters

-autoscale x or y or xy
Autoscale the x axis, y axis, or both axes, overriding the settings for axes scaling and tick mark spacing given in any parameter file specified by the -p option.

Example:

xmgr -autoscale x -p params.par data.dat

Reads parameters from params.par and data from data.dat, but overrides any settings for the X axis found in the parameter file.

-noauto *x* or *y* or *xy*
Suppress autoscaling on the given axis (axes). Use this when a data file has imbedded commands that provide axes scaling and tick mark spacing.

-arrange *rows columns*
Arrange graphs in column major order starting from the lower left corner in a grid *rows* by *columns*. For example, -arrange 2 2 will position the first four graphs (0, 1, 2, 3) in a 2x2 matrix of graphs starting with graph 0 in the lower left corner of the plotting surface. This function is not as complete as the similar function found in Graph/Graph operations/Arrange and gives a sightly different result.

Example: xmgr -arrange 2 2 Creates a 2x2 matrix of graphs.

-rows *rows* Arrange the first rows graphs in *rows* rows.

-cols *columns*
Arrange the first columns graphs in *columns* columns.

-results *results_file* Write the results of regression and anything else that gets written to the results popup to file *results_file*.

-device *device_number* Set the hardcopy device to *device_number*. The device numbering follows:

1 .. PostScript landscape
2 .. PostScript portrait
3 .. FrameMaker .mif landscape
4 .. FrameMaker .mif portrait
5 .. HPGL landscape
6 .. HPGL portrait
7 .. Interleaf landscape
8 .. Interleaf portrait

Example:

xmgr -device 2

Sets the hardcopy device to PostScript portrait.

-eps Tell the PostScript driver to produce EPS.

-printfile *graphics_output_file*
    Write the graphics output to file *graphics_output_file* rather than spool graphics data to a printer.

-fixed *width height*
-portrait
-landscape
-free
    Set the dimensions of the drawing area. Presently, there is no connection between the hardcopy drivers and these settings. This means that the printer needs to be set to conform to the dimensions of the drawing area.

        -fixed *width height* - set the drawing area to have dimensions width and height
        -portrait - set the drawing area to be 8.5x11 in portrait.
        -landscape - set the drawing area to be 8.5x11 in landscape.
        -free - allow the drawing area to be resized (no scrollbars will appear).
-batch *batch_file*
    Execute the commands in *batch_file* on startup. This command line switch is not the same as executing ACE/gr in batch mode as grbatch.

-noask
    Assume the answer is yes to all requests. This means that files will be overwritten without asking if they exist, and, if **Exit** is selected in the GUI, ACE/gr will exit abruptly. This option can be used to overwrite existing files in batch mode. Be careful.

-graph *graph_number*
    Make *graph_number* the current graph. *Graph_number* is an integer between 0 and maxgraph-1 (by default maxgraph, the maximum number of graphs, is 10).

For example, to activate four graphs and read data to each graph:

xmgr -arrange 2 2 data0.dat -graph 1 data1.dat -graph 2 data2.dat -graph 3 data3.dat

3

-graphtype *graph_type* Set the current graph type to graph_type. Where graph_type is one of:

> *xy* linear scaling along both x and y.
> *logx* linear scaling in y, log scaling in x.
> *logy* linear scaling in x, log scaling in y.
> *logxy* log scaling in both x and y.
> *bar* vertical bar chart.
> *hbar* horizontal bar chart.
> *stackedbar* vertical stacked bar chart.
> *stackedhbar* horizontal stacked bar chart.

-type *data_set_type* The format of the next data source is *data_set_type*. Where *data_set_type* is one of:

> *xydx* (XY data with error bars along X)
> *xydy* (XY data with error bars along Y)
> *xydxdx* (XY data with error bars along X of differing values)
> *xydydy* (XY data with error bars along Y of differing values)
> *xydxdy* (error bars along X and Y) xyr (a circle drawn at XY of radius R)
> *xyhilo* (XY data with high-low open-close data)

This setting remains in effect until a new format type is specified.

-ihl *IHL_formatted_file*
> Assume IHL_formatted_file or the next source of data is in IHL format (a local format).

-nxy
> The format of the next data source is X1, Y1, Y2, ..., Yn.

-xydx
> The format of the next data source is X Y DX

-xydy
> The format of the next data source is X Y DY

-xydxdx
> The format of the next data source is X Y DX1 DX2

-xydydy
> The format of the next data source is X Y DY1 DY2

-xydxdy
> The format of the next data source is X Y DX DY

-xyz
> The format of the next data source is X Y Z where Z is a value drawn as text at X, Y.

-xyr
> The format of the next data source is X Y R, where a circle of radius R is drawn at X, Y.

4

-xybox

The format of the next data source is XMIN YMIN XMAX YMAX VALUE, where (XMIN, YMIN) is the lower left-hand corner of a rectangle and (XMAX, YMAX) are the upper right-hand corner of a rectangle. VALUE is an integer used to specify the color to use to fill the rectangle. To set the fill, use Graph/Symbols and set the fill style to anything but None.

-boxplot

The format of the next data source is X AVG

-ihl

The format of the next data source is in IHL format, this format leads to a data set of type XY.

-hilo

The format of the next data source is X HIGH LOW OPEN CLOSE, where a symbol denoting the open, close, the high and the low values for an observation is drawn at X.

-netcdf *netCDF file*
-netcdfxy [*xvar* or "null"] *yvar*

Where *xvar* and *yvar* are the names of the variables to read. If *xvar* == "null", then load the index of *yvar*. These command line options may be used to read variables from a netCDF file. For example:

xmgr -netcdf t.nc -netcdfxy null y

Reads y from t.nc and loads to X the index (starting from 1).

xmgr -netcdf t.nc -netcdfxy x y -netcdfxy x1 y1

Reads x and y from t.nc, then x1 and y1 from the same file. 2 sets are created, both of type XY, one with (x, y), the other composed of (x1, y1). Another example reading data from 2 CDF files:

xmgr -netcdf t.nc -netcdfxy x y -netcdf t2.nc -netcdfxy x1 y1

-saveall *filename*

Write all active sets to *filename*.

-wd

Set the current working directory. This option can be used to read files from several different directories. For example:

xmgr -wd examples mlo.dat -wd ../moredat brw.d hilo.dat

Unfortunately, this option will not allow wild cards, as they are expanded in the current directory only.

-image *X_window_dump_file*

Read the window dump file *X_window_dump_file*. The format is the same as generated by xwd or Frame's capture utility.

-imagexy *X Y*

Set the upper left-hand corner position of the image at screen coordinates (X, Y). Note that screen coordinates have as their origin the upper left-hand corner of the drawing area and positive, increasing Y points down.

-block

Assume the format of the next data file is block data. No sets are formed by reading block data, and only one set of block data is allowed in an active session. To create sets from the block data, use **Edit/Block data.**

-bxy *col1:col2*

Create sets from a previously read block data. Using -bxy col1:col2, will create a set using the current set type (default is XY) and using column *col1* and column *col2* from the most recently read set of block data. *Col1* and *col2* are integers indicating the numbers of the columns - column numbers start from 1. For example:

xmgr -block block.d -bxy 1:4 -bxy 2:6

will read block.d and generate 2 sets of type XY using columns 1 and 4 for the first set, and 2 and 6 for the second set.

xmgr -block block.d -type xydy -bxy 1:3:4

will read block.d and generate a set of type XYDY (an XY set with error bars) using column 1 and 3 for X and Y, and column 4 for the error bars.

-log *x* or *y* or *xy*

Set the current graph type to logarithmic on axis *x* or *y* or both axes by *xy*.

-parameter *parameter_file*
-p *parameter_file*

Read the parameter file parameter_file.

-pexec *parameter_string*

Interpret *parameter_string* as a parameter setting.

-result *result_file*

Write results from regression to *result_file*.

-rvideo

Exchange the color indices for black and white.

-mono

Limit colors to black and white, affects the display only.

-dc

Allow double click canvas operations, overriding any application default setting.

-nodc
> Disallow double click canvas operations.

-redraw
> Redraw the entire page if ACE/gr receives an Expose event.

-noredraw
> Do not redraw the entire page when Expose events are received.

-maxcolors *number_of_colors*
> Set the maximum number of colors allocated.

-world *xmin ymin xmax ymax*
> Set the scaling of the axes for the current graph. The X axis will run from (xmin, xmax) the Y axis from (ymin, ymax)

-view *xmin ymin xmax ymax*
> Set the viewport of the current graph to a rectangle described by (xmin, ymin) and (xmax, ymax).

-source *data_source*
> Set source for the next data file, the argument takes on the following values:

> disk - Data file is on disk.
> pipe - Read from a pipe.
> stdin - ACE/gr is in a pipe, read from standard input.

The source setting stays in effect until the next -source command is given.

-pipe
> Read from standard input and plot each set when the end of the set is seen. Use the -pipe option to monitor the results coming from a model or other data stream.

-remove
> Erase the most recently read file. Use this option when using ACE/gr to read a temporarily constructed file that is not needed once read. Note that this option does not request confirmation.

-legend *load*
> For each data set read, make the set legend label the same as the filename the data set originated.

-seed *seed_value*
> Initialize the random number generator using seed *seed_value*. This value is used in a call to the math library function srand48().

-GXinvert
-GXxor
> GXinvert and GXxor set the type of graphics operation used to draw rubberband lines and draw the graph focus markers. In a color mapped display, the xor of a source and destination pixel may not generate a color visible on the drawing area. The default is to use xor to draw rubberband lines, if the lines don't show, try -GXinvert. This draw mode can be set via X resource settings (see below).

7

-maxplot *number_of_sets*

    Set the maximum number of data sets per graph. *Number_of_sets* must be greater than 30. If -maxplot is used in conjunction with -maxgraph, then the -maxplot setting must precede the -maxgraph setting.

-maxgraph *number_of_graphs*

    Set the maximum number of graphs per session. *Number_of_graphs* must be greater than 10. If -maxgraph is used in conjunction with -maxplot, then -maxplot must precede the use of -maxgraph.

-nonl

    Activate the Non-linear curve fitting module. WARNING, this module is a prototype and should not be used for analysis.

-digit

    Activate the Digitizing module. WARNING, this module is a prototype and should not be used for analysis.

-usage

    Display a brief explanation of command line settings

data_files

    The files or commands in the case of a pipe from which data are read.

2.3 Setting ACE/gr Defaults

On startup ACE/gr will look for a .xmgrrc file in the users' home directory allowing changes to the default behavior of ACE/gr. The construction of the ACE/gr default file is the same as for a parameter file.

2.4 Setting ACE/gr X Defaults

The following X resource settings are supported:

Xmgr.invertDraw: *False*

    Use GXinvert to draw rubberband lines and the graph focus markers if this resource is set to True.

Xmgr.reverseVideo: *False*

    A boolean resource used to exchange the color indices for black and white.

Xmgr.maxSets: 30

    Set the number of sets per graph.

Xmgr.maxGraphs: 10

    Set the number of graphs.

Xmgr.maxColors: 17

    Set the number of colors allocated. There is no support in the graphical portion of the user interface for more than 16.

Xmgr.verifyAction: No

    Applies to the Pick set operations and allows an opportunity to accept or cancel any operation performed.

Xmgr.allowDoubleClick: Yes

    When Yes, allow double clicks on the canvas to bring up various popups depending on the location of the pointer when the double click occurs. Double clicking to the left of a graph brings up the **Graph-Tick labels/Tick marks...** popup set to the Y-axis, below the graph brings up the **Graph-Tick labels/Tick marks...** pop up set to the X-axis, inside the graph brings up the **File-Read sets...** popup if no sets are active or the **Graph-Symbols...** popup set to the set nearest the pointer. Double clicking above the graph brings up the **Graph-Titles...** popup, to the right of the graph, the **Graph-Legends...** popup. The command line option to set this resource is -dc or -nodc.

Xmgr.autoscaleOnRead: No

    When this resource is Yes or True, allow autoscaling to occur each time a set is read from the File/Read sets popup.

Xmgr.backingstore: No

    Have ACE/gr handle backing store if the server doesn't. The command line option for this is -bs or -nobs.

2.5 Environment Variables

GR_HOME

    Set the location of ACE/gr. This will be where help files, auxiliary programs, and examples are located. If you are unable to find the location of this directory, contact your system administrator.

GR_PS_PRSTR

    Set the command string to use for spooling to a PostScript printer. The default (unless changed during the build process) is "lpr -h".

GR_MIF_PRSTR

    Set the command string to use for printing the FrameMaker .mif format. The default (unless changed during the build process) is "cat >acegr.mif < ", i.e., output is to a file.

GR_HPGL_PRSTR

    Set the command string to use for spooling to an HPGL capable printer or plotter.

GR_HDEV

    Select the hardcopy device.

The device numbering is as follows:

    1. PostScript landscape
    2. PostScript portrait
    3. FrameMaker interchange format (.mif) landscape
    4. FrameMaker interchange format (.mif) portrait.
    5. HPGL landscape
    6. HPGL portrait

7. Interleaf landscape
8. Interleaf portrait

2.6 Batch Printing

Executing ACE/gr as grbatch suppresses the initialization of the graphical user interface and allows hardcopy plots to be made without intervention.

2.7 Fonts

To change fonts within a string, precede the font number listed below by a backslash. To turn subscripting or superscripting on or off, use \s for subscripts and \S for superscripting. Font selection is current for the remainder of the string or until the next font change. Subscripts and superscripts remain for the remainder of the string or until \N is seen. To print a backslash, use \\. To backspace, use \b. To begin underlining, use \u, to stop underlining, use \U. \+ increases the size of the characters and \- decreases the size of the characters.

ACE/gr uses the Hershey fonts to draw text on the screen, but PostScript fonts for hardcopy. There are discrepancies between the two sets of fonts and, of this writing, there are problems with the mapping of Greek and special characters.

Font # Font
0 .... Complex Roman on screen, maps to Times-Roman.
1 .... Triplex Roman on screen, maps to Times-Bold.
2 .... Complex Italic on screen, maps to Times-Italic.
3 .... Triplex Italic on screen, maps to Times-Bold-Italic.
4 .... Simplex on screen, maps to Helvetica.
5 .... Duplex on screen, maps to Helvetica-Bold.
6 .... Complex Italic on screen, maps to Helvetica-Oblique.
7 .... Triplex Italic on screen, maps to Helvetica-Bold-Oblique.
8 .... Simplex Greek on screen, maps to the lower 128 characters of the Symbol font.
9 .... Symbols on screen, maps to the upper 128 characters of the Symbol font.
x .... Special symbols

Summary of other special commands
       + ... increase size
       - .... decrease size
       b .... backspace length of previous character
       s .... begin subscripting
       S .... begin superscripting
       u .... begin underline
       U .... stop underline
       N .... return to normal

Example:\0F\sX\N(\8e\0) = sin(\8e\0)*e\S(-X)\N*cos(\8e\0)

prints roughly FX(e) = sin(e)*e -X*cos(e) using font 0 and e prints as epsilon from the Simplex Greek font.

NOTE: Special characters are mapped to the keyboard, the present mapping is not very well organized and may change.

3.0 GUIDE TO MENUS AND POPUPS

3.1 Main Panel

There are 5 main areas in the display, the menu bar at the top, the locator bar just below, the tool bar along the lefthand side of the display, the status bar at the bottom, and the drawing area where graphics are drawn. The following items describe the locator bar, the toolbar, the status bar and hot keys for ACE/gr. Sections 3.2-3.5 describe the pull down options and corresponding menus where applicable.

3.1.1 Tool Bar

The toolbar runs along the lefthand side of the display, and provides functions to manipulate the axes scaling of active graphs, by scrolling and mouse powered zooms.

3.1.1.1 Draw. Click on **Draw** to freshen up the drawing area with the current set of graphs, parameters and active sets. Most operations perform an automatic redraw, this may be undesirable in the cases where the sets are large or there are several active graphs. By playing with the options in **Graph/ Draw options** and using the **Draw** button, much of the tedium associated with slow redraw rates can be alleviated. If you feel something should have happened after performing an operation, press this button to force a draw operation to freshen up the display. If the X display being used does not do backing store, then **Draw** will give a clean picture if the graphics get damaged when overlaid by popups or other applications.

3.1.1.2 Zoom. Click on Zoom to zoom in on the plot. Then specify the zoom area by selecting one corner and then the other.

3.1.1.3 Autoscale. Click on Autoscale to generate a default scaling for both the X and Y axes in the current graph.

3.1.1.4 Expand. Click on Expand to incrementally zoom in on the data.

3.1.1.5 Shrink. Click on Shrink to incrementally zoom out from the data.

3.1.1.6 Page Left. Click on Left to incrementally shift the axes to the left.

3.1.1.7 Page Right. Click on Right to incrementally shift the axes to the right.

3.1.1.8 Page Down. Click on Down to incrementally shift the axes down.

3.1.1.9 Page Up. Click on Up to incrementally shift the axes up.

3.1.1.10 Auto Ticks. Using any of the buttons that affect the scale of the axes, it is likely that unpleasing tick marks will result. By clicking on AutoT, the tick spacing will be set using the current autoscaling parameters (set in Graph/Autoscale).

3.1.1.11 Auto Scale on a Picked Set. Click on AutoO then click near a point in the set of interest to autoscale the current graph with respect to the selected sets bounding box.

3.1.1.12 Zoom X. Select a Zoom line used to scale the X axis of the current graph.

3.1.1.13 Zoom Y. Select a Zoom line used to scale the Y axis of the current graph.

3.1.1.14 Auto X. Autoscale the X axis only.

3.1.1.15 Auto Y. Autoscale the Y axis only.

3.1.1.16 Push 'n Zoom. Save the current graph's axes scales and tick mark settings, and prepare for a zoom. This function can be used to zoom in on an area of interest, then restore the previous graph scale.

3.1.1.17 Push. Save the current graph's axes scales and tick mark settings for later retrieval. Use Pop to restore.

3.1.1.18 Pop. Restore the current graph's axes scale saved from the most recent use of Push.

3.1.1.19 Cycle. Cycle allows the contents of the current graph's world stack to be displayed in stack order. The contents of the stack are unaffected.

3.1.1.20 Stack Depth. Report on the current depth of the stack. Each graph has its own stack.

3.1.1.21 Current World. When cycling through the current graph's world stack, display the position in the stack.

3.1.1.22 Exit. Exit ACE/gr.

3.1.2 Locator Bar

The Locator bar, found directly under the main menu bar, displays the current graph and the position of the pointer. To set properties of the locator display text, see View/Locator props.

3.1.3 Status Bar

The Status bar, at the bottom of the display, reports on items of interest and provides a line of help when the mouse is used in any operation.

3.1.4 Hot Keys

When the mouse is on the canvas (where the graph is drawn), there are some shortcuts that can be taken to bring up several popups. They are:

    ^A - autoscale plot
    ^B - draw a box (using the current settings for boxes)
    ^D - delete an object (string, line, or box)
    ^N - move an object (string, line, or box)
    ^P - draw a line (using the current settings for lines)
    ^V - set the viewport with the mouse
    ^W - write a string (using the current settings for strings)
    ^X - exit ACE/gr
    ^Z - enable zoom

12

3.2 File

Read data from disk or pipes, read variables from netCDF or HDF files, read and write plot parameters, read block data, write one or more ACE/gr data sets, set the current working directory, set the printer options, print (hardcopy), access the command line interpreter, describe the status of data sets, graphs and regions, and display results of computations and other operations.

3.2.1 Read Sets

Read one or more data sets. The list item at the top of the popup displays the contents of the current directory. Select a file or directory by clicking on the item with the right mouse button. The selection is placed in the text item labeled "File:" and checked to see if it is a directory or a file. Selecting a directory causes ACE/gr to change to the new directory and reload the file list with the contents of the new directory. Selecting a file does nothing until the "Accept" button is pressed or is entered. Before "Accepting the file", the items describing the file type, file source, and the graph in which to read the data need to be properly set. The type of data file can be one of several formats:

A 2 column multi-data set file. Sets are separated by a line containing non-numeric characters, that are neither comment lines (lines with a "#" in column 1) nor parameter lines (lines beginning with an "@"). The sets generated from this file type are all of type XY.

A multi column data file. X is assumed to be in the first column, and Y1, Y2, ..., up to Y30 in the remaining columns. The sets generated with this data file format are all of type XY.

IHL format. A 3 column data file with the first line an integer value with the number of points to follow.

Binary format. - not defined as of this writing.

X Y DX format. A 3 column data file consisting of X, Y and a quantity used to form an error bar parallel to the X-axis, i.e., the point will be plotted with an error bar at (X+-DX, Y).

X Y DY format. A 3 column data file consisting of X, Y and a quantity used to form an error bar parallel to the Y-axis, i.e., the point will be plotted with an error bar at (X, Y+-DY).

X Y DX1 DX2 format. A 4 column data file consisting of X, Y, and the errors in X. The error bar riser is drawn from (X+DX1, Y) to(X-DX2, Y). If both DX1 and DX2 are >0 then the error bar will bracket the datum. If DX2 is <0 and DX1 > 0 the error bars are drawn to the right of the datum. Likewise, if DX1 < 0 and DX2 > 0 the error bar is drawn to the left of the datum. There are 4 cases for the error bars' position with respect to the datum:

    1. DX1 > 0 and DX2 > 0 ... Error bars bracket the datum.
    2. DX1 > 0 and DX2 < 0 ... Error bars are drawn to the right of the datum.
    3. DX1 < 0 and DX2 > 0 ... Error bars are drawn to the left of the datum.
    4. DX1 < 0 and DX2 < 0 ... Error bars bracket the datum, but reversed from case 1.

X Y DY1 DY2 format. A 4 column data file consisting of X, Y, and the errors in Y. The error bar riser is drawn from (X, Y+DY1) to (X,Y-DY2). If both DY1 and DY2 are >0 then the error bar will bracket the datum. If DY2 is <0 and DY1 > 0 the error bars are drawn to the right of the datum. Likewise, if

DY1 < 0 and DY2 > 0 the error bar is drawn to the left of the datum. There are 4 cases for the error bars' position with respect to the datum:

1. DY1 > 0 and DY2 > 0 ... Error bars bracket the datum.
2. DY1 > 0 and DY2 < 0 ... Error bars are drawn above the datum.
3. DY1 < 0 and DY2 > 0 ... Error bars are drawn below the datum.
4. DY1 < 0 and DY2 < 0 ... Error bars bracket the datum, but reversed from case 1.

X Y DX DY format. A 4 column data file consisting of X, Y, and the errors in X and Y. The error bar risers are drawn from (X+DX, Y) to(X-DX, Y) and (X, Y+DY) to (X, Y-DY).

X Y Z format. A 3 column data file consisting of X, Y, Z. Sets with this type are drawn with the Z value in text at (X, Y).

X Y R format. A 3 column data file consisting of X, Y, R. Sets with this type are drawn with a circle of radius R at (X, Y).

X HI LO OPEN CLOSE format. A 5 column data file consisting of the high, low, open and close values for an observation at X. A data set of this type uses the line style, line width and line color for a normal set when the symbol is drawn.

Read from: sets the source of the data, either a disk file or a pipe. In the case of a pipe, the information provided by the file filter item is not used, as the "file" in this case will be a command as typed at the UNIX prompt.

Read to graph: selects a particular graph to serve as the repository for the incoming data. The "Current" graph is the graph that has the focus, and can be any of the available graphs. If you aren't sure which graph is current, the locator item on the main panel displays the current graph number.

Selecting **Autoscale on read**: forces an autoscale of the target graph each time a set is read.

Note: Data are read into the next available set. All data are assumed to be delimited by tabs or blanks.

For the xy, xydx, xydy, xydxdx, xydydy, xydxdy, xyz, xyr data file formats, sets are separated by a line containing non-numeric characters, that are neither comment lines (lines with a "#" in column 1) nor parameter lines (lines beginning with an "@"). Data files can have imbedded comments by placing a "#" in the first column of the comment line. Plot parameters can be set by placing a "@" in the first column of the line followed immediately by the parameter name and setting. See the command line reference for a description of plot parameters and other commands that may be placed in a data file using the "@" character. Comment lines and parameter lines can occur at any place in the data file, and are not used to indicate data set separators. I recommend a single "&" on a line to use as a data set separator.

Press **Accept** to read the data. If the read was successful, the graph of the data is drawn using the current plot scaling parameters. If the data lies outside the current plotting limits, the new set(s) will not be visible until the plotting limits are changed by autoscaling or manually through the View/Define world popup.

Press **Cancel** to close the popup.

Open the **Status** popup to get information on the state of sets. If reading the file generates more than 10 errors, you'll be prompted by an alert requesting whether or not you'd like to continue. These errors generally arise when an attempt is made to read a text file - ACE/gr relies on the interpretive capabilities of sscanf() to parse the data. ACE/gr will read data sets until there are no more sets available, after the maximum number of sets have been used, you'll need to free some sets using the [Kill] or [Kill all] items in the Edit/Set operations pull right.

3.2.2 Read netCDF

File/Read netCDF
Select X: Select Y:
netCDF file:
Load to set:
Autoscale on read
Accept Files... Update Query Done

Enter the name of the netCDF file in the text item labeled netCDF file: then click on "Update" (note that if -netcdf [filename] was used on the command line, this step is not needed, as the lists will be loaded automatically). This will read the names of all one dimensional variables into the two scrolled lists. The leftmost list is for X and the rightmost for Y. There is an additional item in the X list called INDEX (I'm hoping here that no one names a variable INDEX, maybe I need to change this) that indicates that for X, use the index of the Y variable (numbering starts from 1).

Select on a variable from the X list to use for X and select a variable from the Y list (only single selections for now).

If information about the selected variables is desired, click on Query.

Select a set using "Load to set:" - the default is to load to the next available set.

If things look OK, click on **Accept** to read the data and load the set. Use "Autoscale on read" to set/unset autoscaling after a successful read.

To read data from a new CDF file, enter the filename, or use File... to scan the directory. Click on Update to freshen up the X and Y selection lists.

Note: Presently, only sets of type XY are created.

Note: ACE/gr scans and replaces commas with blanks and the "D" in FORTRAN double precision exponential formats with "e".

3.2.3 Read Parameters

Enter the parameter file name on the line labeled "File:", press "Accept" to read a saved state of adjustable parameters, legends, strings, etc., to the graph selected by the "Read to graph" cycle. As the graph number is saved in the parameter file this isn't used at the present time.

3.2.4 Read Block Data

Select the data source, either Disk or Pipe and enter the block data file name in the text item labeled "File:" to read a data set in block data file format. The block data file format consists of columns of data. For example, the following is a block data file consisting of 4 columns:

```
 1  2  3  4  5
 6  7  8  9 10
11 12 13 14 15
```

After reading the file of block data, the popup will close if no errors were found, and the Edit/Block data popup will take its place. Use the functions in the Edit/Block data popup to create sets from the block data.

NOTE: All items are assumed delimited by spaces or tabs. There is presently a 30 column limit. Only one set of block data per session, i.e., reading another set of block data will obliterate the previous set of block data.

3.2.5 Write Sets

Write one or all sets to disk. Select the set to write using the cycle displaying the set numbers (there is an item for selecting all active sets). The item denoted "Format" is the C language format string that will be used to format the data for output - the default should work well for most cases.

Enter the file name on the line labeled "File:" and press the button marked [Accept], to write the data. The item, "Embed parameters," causes ACE/gr to write not only the data, but the parameters describing the graph or graphs as well.

A complete dump of the contents of ACE/gr may be accomplished by selecting all sets, embed parameters, and all active graphs. The file generated this way can then be read as a normal data file, and should recreate the originating environment.

3.2.6 Write Parameters

Select the graph from which to save the parameters. This can be the current graph, any particular graph, or all active graphs. Enter the parameter file name on the line labeled "Write parameters to", press "Accept" to write the current state of adjustable parameters, legends, strings, etc. A prompt will appear if the action would overwrite an existing file, allowing the operation to be canceled.

3.2.7 Save All

Perform a "Write sets" using all active sets, all active graphs, and embed parameters. Use this to save the current session.

3.2.8 Clear All

Kill all sets, graphs, and annotative text, lines, and boxes. After performing this action, it will be necessary to activate a graph using View/Graphs/Activate graphs.

16

3.2.9 Print

Generate a hardcopy of the current plot on the device specified in the File/Printer setup popup.

3.2.10 Printer Setup

Set the hardcopy device parameters. Select the device, where to print (either the printer or a file), and the string to use as a spooling command for the print job, or file name if output is directed to disk. Click on the "Accept" button to make the selections above current, or "Done" to close the popup and cancel the operation. If the selection is accepted, the next time File/Print is selected, hardcopy output will be to the selected device. Click on the "Print" button to accept the settings and print a hardcopy.

3.2.11 Command Interpreter

Command driven version of the interface to ACE/gr. Here, commands are typed at the "Command:" text item and executed when is entered. The command will be parsed and executed, and if no errors are found, the command line is placed in the history list. Items in the history list can be recalled by simply clicking on them with the left mouse button. Save a history list by clicking on the "Save..." button popping up the "Save commands" popup. Clicking on "Read..." will open the "Read commands" popup and allows previously stored commands to be loaded into the history list for future access. "Clear" empties the history list. "Replay" cycles through each item in the history list and executes each instruction in the list. See the command line reference for a description of the command line syntax.

3.2.12 Status

The **Status** popup displays useful information about the state of sets, regions, and graphs. In addition to this, several useful operations are provided to manipulate sets and graphs. At the bottom of the popup are the controls for displaying what appears in the top portion of the popup. Directly above this is a panel that provides operations on the objects displayed in the scrolled region.

Starting with the bottom panel, from left to right, the selections are:

Close   Close the Status popup.

Update   Freshen up the status display.

Write   Write the contents of the status display to the Results popup, from there, the information provided by Status can be written to disk.

Page+   Page forward through the status display. There is a limit as to how many sets can be displayed in the Status popup at one time. By default, the maximum is 30. When the number of sets is less than 30, this button is not needed; use the scroll bars to view the status of the sets that are not shown in the display.

Page-   Page backward through the status display. There is a limit as to how many sets can be displayed in the Status display region at one time. By default, the maximum is 30. When the number of sets is less than 30, this button is not needed; use the scroll bars to view the status of the sets that are not shown in the display.

Home   Go to the first page of the status display.

End   Go to the last page of the status display.

Display   An option menu used to select the object types to view, sets, graphs, or regions. The operations panel directly above will change as the object type in this item is changed.

Operations in the **Status** popup are performed by clicking on the operation, then clicking on one or more set numbers shown in the display. There is a status line provided that will give a bit of help on what actions to take to perform a given operation. Generally, the cursor will change to reflect the nature of the operation.

When the selection for **Display** is **Sets**, the operations provided are:

Kill   Clicking on Kill activates the pointer such that clicking on a set number shown in the display of the Status popup kills the set. After a set is Killed, the point remains hot, the Kill operation can be performed again, until the Cancel button is pressed.

Deact   Clicking on Deact followed by clicking on a set number in the scrolled window will deactivate the set (deactivation merely hides the set, the data is not released). The function can be useful when there are lots of sets being drawn, and the interest is in a subset of the sets being display in the drawing area.

React   Clicking on React followed by clicking on a set number in the scrolled window will reactivate the set if it has previously be deactivated.

Copy   Clicking on Copy followed by clicking on a set number in the scrolled window will select the set to copy from. Click again on another set number to perform the copy. One trick that can be performed here is that this operation can be used to copy sets from one graph to another by changing the current graph. The current graph must be changed by either clicking on the graph to receive the focus on the drawing area, or using Graph/Graph operations.

Move   Clicking on Move followed by clicking on a set number in the scrolled window will select the set to move. Click again on another set number to perform the move. One trick that can be performed here is that this operation can be used to move sets from one graph to another by changing the current graph. The current graph must be changed by either clicking on the graph to receive the focus on the drawing area, or using Graph/Graph operations.

Auto   Clicking on Auto followed by clicking on a set number in the scrolled window will select autoscale the graph axes such that the set in its entirety is displayed in the drawing area.

Reverse   Clicking on Reverse followed by clicking on a set number in the scrolled window will select reverse the order of the points in the set.

Join   Clicking on Join followed by clicking on a set number in the scrolled window will select the set to join to, Click again on another set number to complete the join.

Pack   Clicking on Pack renumbers sets such that there are no gaps in the set structure. If the active sets are 0, 3, 5, following the pack operation, the sets will be numbered 0, 1, 2.

Cancel   Clicking on Cancel cancels any of the operations above that require more than a single set to function (Copy, Move, Join). Also, most of the operations stay in effect until Cancel is pressed.

When the selection for **Display** is **Graphs**, the operations provided are:

Activate   Clicking on Activate then clicking on a graph number activates the graph.

Copy   Clicking on Copy then clicking on a graph number selects the graph to copy from. Clicking on another graph selects the graph to copy to.

Move   Clicking on Move then clicking on a graph number selects the graph to move from. Clicking on another graph selects the graph to move to.

Swap   To swap 2 graphs, click on Swap then click on a graph number to select the first graph, then click again on the second graph to perform the swap.

Hide   To hide a graph, click on Hide then click on a graph number.

Show   To undo the effect of Hide, click on Show then click on a graph number.

Focus   To set the graph focus (the graph with the focus is referred to as the current graph), click on Focus then click on a graph number.

Kill   To kill a graph, click on Kill then click on a graph number.

Cancel   Cancel operations that require 2 selections and take the pointer out of whatever operation was previously selected.

When the selection for **Display** is **Regions**, no operations are provided at the present time.

3.2.13 Results

Display the results of the regression routine and the output of the "Write" button in File/ Status in the monitor window.

3.2.14 Exit

Click on the Exit button to terminate the session with ACE/gr.

3.3 Data

3.3.1 Status

See Section 3.1.12 Status (under File menu).

3.3.2 Results

See Section 3.1.13 Results (under File menu).

3.3.3 Transformations

Some of these operations generate new sets, if you run out of sets use Edit/Set operations (described below) to kill unneeded sets. Also, the scaling of the world coordinate system may be inappropriate for the results of many of these operations. Use the Status popup to deter mine the appropriate scaling factors to use in "Define world" (above), or use "Autoscale..." (in View/Autoscale above). Most functions operate on active sets only.

3.3.3.1 Evaluate Expressions. Evaluates a formula defined in infix fashion.

Select the set on which the formula will operate, (set must be active, use File/Status to find the current state of sets). If you desire, the result can be loaded to a new set rather than overwriting the set used for computations, use the panel cycle denoted "Result to" to inform ACE/gr you'd like the result placed in a new set (if there is one). A new set is created only if a single set is selected from step 1. If "All sets" are selected then the results will overwrite all active sets.

Enter the formula, the syntax is:

(x,y,a,b,c,d)=

where (x,y,a,b,c,d) are defined as the x and y of the currently selected set and a,b,c,d are scratch arrays that can be used to perform operations between sets. Case is ignored, so X=COS(X) is correct.

Click on **Accept** when you are satisfied with everything.

Variables:
      a == reference to scratch array
      b == reference to scratch array
      c == reference to scratch array
      d == reference to scratch array

Functions:
      abs(x) == absolute value
      acos(x) == arccosine
      asin(x) == arcsine
      atan(x) == arctangent
      atan2(y,x) == FORTRAN ATAN2
      ceil(x) == greatest integer function
      cos(x) == cosine
      deg == 180.0/PI
      dx == span of world coordinate system in x
      dy == span of world coordinate system in y
      erf(x) == error function
      erfc(x) == complement of error function
      exp(x) == e^x
      floor(x) == least integer function
      index == the index of the current point in the selected set
      int(x) == truncation
      invn(p) == inverse of standard normal (p in [0,1])

invt(p,id) == inverse of Student's t with id degrees of freedom
irand(n) == random integer less than n
gamma(x) == log of gamma function
ln(x) == natural log (should be log)
log(x) == log base 10 (should be log10)
max(x,y) == returns greater of x and y
min(x,y) == returns lesser of x and y
mod(x,y) == mod function (also x % y)
norm(x) == gaussian density function
ormp(x) == cumulative gaussian density (-inf,x)
pi == constant PI
rad == PI/180.0
rand == pseudo random number distributed uniformly on (0.0,1.0)
rnorm(xbar,s) == pseudo random number distributed N(xbar,s)
sin(x) == sine function
sqr(x) == x^2
sqrt(x) == x^0.5
tan(x) == tangent function
x == currently selected set X
y == currently selected set Y

Note: See pars.yacc for the yacc grammar.

Examples:

y=-y
y=x*cos(2*x*PI/100)+sqr(x)
x=(index>10)*(x-5)+(index<=10)*x27

If the index of the current point is greater than 10, then x=x-5 else x=x. Conditionals evaluate to 0 if false, anything else is true.

a=y

Store y of the current set into scratch array "a" you may now select another set and perform the operation y=some function of(a)

3.3.3.2 Load Values. Load a sequence to (x,y) or (a,b,c,d).

3.3.3.3 Load and Evaluate. Evaluate parametric functions.

Enter the functions to be used to define X and Y. Select the independent variable (x,y,a,b,c,d). Enter the start, stop and the number of points items. Press the button **Apply** to evaluate the functions and load the result to a new set.

3.3.3.4 Histogram. Compute a frequency histogram.

Select the set. Enter the width of a bin (all bin widths are the same). Enter the minimum and maximum values of the portion of the data you wish histoed (sic). Note that Xmin and Xmax refer to the RANGE (or Y) of the set, not the domain (or X). Press **Accept** to compute the histogram.

3.3.3.5 Fourier. Compute the Discrete Fourier transform.

Select the set. Select the type of data window, the default is the rectangular window in which case the data is transformed unmodified. The data windows are defined as follows:

> None: Use the default rectangular window
> Triangular: $1.0 - |(i-0.5*(N-1))/(0.5*(N-1))|$
> Hanning: $0.5 * [1-\cos(2*pi*i/(N-1))]$
> Welch: $1-((i-0.5*(N-1))/(0.5*(N+1)))^2$
> Hamming: $0.54-0.46*\cos(2*pi*i/(N-1))$
> Blackman: $0.42-0.5*\cos(2*pi*i/(N-1))+0.08*\cos(4*pi*i/(N-1))$
> Parzen: $1.0 - |(i-0.5*(N-1))/(0.5*(N+1))|$

Select the form of the output, magnitude (spectrum), phase, or the coefficients. The spectrum is computed by $\sqrt{x*x + y*y}$ where x, y are the coefficients computed by the DFT or FFT. Only N/2 values (representing frequencies 0 to PI) are loaded to the resulting set.

If the magnitude or phase is selected, then the next item, "X = ", determines what values should be loaded to X. The index runs from 0 to n/2, the frequency is the cyclical ith fourier frequency, the period is the reciprocal of the frequency with the period of the 0th fourier frequency plotted at T+delt, where T is the total length of the data and delt is the sampling interval.

Select transform or inverse transform. Select real or complex data.

If real is selected, then the data to be transformed is assumed to be in Y, X is assumed equally spaced and is ignored. If complex is selected then the real part is assumed to be in X and the imaginary part in Y.

Press DFT (for small data sets whose length is not a power of 2) - or FFT (for data sets whose length is a power of 2).

Click on **Window only** to generate a windowed version of the data in a new set. NOTE: Small for the DFT is < 1000 points. The DFT is $O(N**2)$ and can be quite time consuming to compute for large N.

3.3.3.6 Running Averages. Compute a running average, median, minimum, maximum, or standard deviation.

Select the method. Select the set. Set the length of the running method in the text item marked **Length**, it must be less than the set length. Click on **Accept**.

3.3.3.7 Regression. Perform linear or polynomial regression.

Select the set. Select the degree of fit. Select fitted curve or residuals to load. Press the button marked "Regress". A set is loaded (if there is one) with the resulting curve and a summary of the statistical results are written to the **Result** popup.

3.3.3.8 *Differences*. Numerical differentiation.

Select the set. Select the method - one of forward, backward, or centered difference. Assumes unevenly spaced data, increasing in X. Click on **Accept**. A set is loaded (if there is one) with the resulting curve.

3.3.3.9 *Seasonal Differences*. Difference a set by a given lag.

Select the set. Enter the value for the lag. Assumptions are evenly spaced data, increasing in X. Press the button marked **Accept**. Use **Pick** to use the mouse to select the set by clicking near a point in the set as displayed in the drawing area. A set is loaded (if there is one) with the resulting curve.

3.3.3.10 *Integration*. Numerical integration.

Select the set. Select the form of the results, the item marked cumulative sum will construct a set composed of the current value of the integral at a given X. Sum only reports just the value on the next line. Assumes unevenly spaced data, increasing in X. Press the button marked Integrate. A set is loaded (if there is one) with the resulting curve if cumulative sum is chosen.

3.3.3.11 *X-corr*. Cross/auto-correlation.

Select both sets (use the same set if auto-correlation is desired). Select the lag, N/3 is a reasonable value (your mileage may vary). Select bias - generally this will not make any difference for large data sets with lags << the length of the set, I was just curious. The difference is division by N (biased) or N-lag (unbiased). A set is loaded (if there is one) with the resulting curve.

3.3.3.12 *Spline*. Compute a spline fit to a set.

Select the set. Select the starting value of X for the fitted curve. Select the ending value of X. Select the number of samples. The spline curve will be evaluated at X+i*(Manx - minx)/Steps for each i in (0, Steps-1). Press "Spline". A set is loaded (if there is one) with the resulting curve. Notes: The code to compute the spline is a literal translation of the code in F.M.

3.3.3.13 *Sample*. Sample a set pointwise or by a logical expression.

Select the set. Select the type of sample, either Start/step or Logical expression. If Start/step is selected, then enter the starting index to begin the sample.

Select the number of points to skip between samples in Step. If Logical expression is selected, enter the expression in the text item denoted Exp.:. Values of the expression not equal to zero are interpreted as TRUE, and the point is accepted. Any expression evaluating to zero will result in the point being ignored.

3.3.3.14 *Digital Filter*. Apply a digital filter to a set.

Select the set to be filtered. Select the set with the filter weights.

3.3.3.15 *Linear Convolution*. Perform convolution of 2 sets.

3.3.3.16 *GeometricTtransformations*. Apply rotations, scaling, and translations to a set.

Select the order in which to apply the transformations. Press the button "Apply" to perform the transformation. Restrictions: Only the set X and Y are transformed, additional vectors attached to a set, such as error bars, are not transformed.

3.3.4 Set Operations

Set operations allow sets to be created, destroyed, written to disk, sorted.

3.3.4.1 Pick Operations. Pick Ops uses the point in a set nearest to the location of the pointer when clicked to select the set for the chosen operation. The nearest point is determined by the Euclidian distance which can be very different from the point that is visually closer. As long as there are sets with points, the operation will always be performed, which means that a bit of care should be exercised using these operations.

Kill nearest set - Kill the set nearest the pointer.

Copy nearest set - Select a set to copy with one click, and the destination, which may be in another graph, with another click.

Move nearest set - Select a set to move with one click, and the destination, which may be in another graph, with another click.

Reverse nearest set - Select a set to reverse by clicking near a point in a set. The result is that the order of the points is reversed (point N becomes point 1, and point 1 becomes point number N, etc.).

De-activate nearest set - Select a set to deactivate by clicking near a point in a set. The result is that the set will act as if undefined for most operations, and will not be drawn even if the line or symbol selections would cause it to be drawn. The data is not destroyed, and the set can be reactivated using the functions in File/Status or Data/Set Ops.

Join nearest sets - Click near a point in one set to use for the join, then click again near a point in another set to select the second set. The second set will be appended to the end of the first set.

Delete range in nearest set - Click near a point in one set to use for the start of the range to delete, then again near the point at the end of the range to delete. The result is that all points between the 2 selected points will be removed from the set, including the 2 points selected.

Cancel operation - Cancel any operation in progress.

The remaining items in the Set Ops toolbox allow sets to be created, destroyed, written to disk, moved, sorted, deactivated, reactivated.

3.3.4.2 Activate. Make a set active and able to participate in operations. To activate a set, select the set number, specify the number of points in the set and click on Accept.

3.3.4.3 Deactivate. Make a set inactive and unable to participate in operations. The data associated with a deactivated set are still available and all plot parameters associated with the set are unchanged. Use the Re-activate set item below to make the set known to ACE/gr again. This item is include for those situations where a set is to be ignored temporarily, but needed later in the session.

Select the set to Deactivate. Press the button marked **Accept**.

3.3.4.4 Re-activate. Undo the effect of Deactivate.

Select the set to Reactivate, it is a no-op to Reactivate an unused set. Press the button marked **Accept**.

3.3.4.5 Set Length. Set the length of a set.

Select the set. Fill in the item marked Length. Press the button marked **Accept**.

3.3.4.6 Change Set Type. Set the type of a set.

Select the set. Select the type to set. Press the button marked **Accept**.

3.3.4.7 Copy. Copy one set to another in a possibly different graph.

Select the set to copy from. Select the set to copy to. Select the graph to receive the copy. Press the button marked **Accept**.

3.3.4.8 Move. Move one set to another in a possibly different graph. Select the set to move from. Select the set to move to. Select the graph to receive the set. Press the button marked **Accept**.

3.3.4.9 Drop Points. Drop points from a set.

Select the set. Fill in the items "Start drop" and "End drop". Press the button marked **Accept**. NOTE: Points in a set are numbered from 1 to N, where N is the number of points in the set (as opposed to sets which are numbered from 0).

3.3.4.10 Join. Merge 2 sets together.

Select the set that will be appended. Select the set accepting the previous set. Press the button marked **Accept** to append the first set to the second.

3.3.4.11 Split. Divide a set into other sets.

Select the set to split. Enter the length of the resulting sets (say N). Press the button marked **Accept** to divide the selected set into (length of selected set) / N sets with the remainder to the last set.

The first set will be the set selected to split and will contain the first N points.

3.3.4.12 Kill. Eliminate a set.

Select the set to kill or All to kill every active set. To save the parameter settings, toggle "Preserve parameters." Press the button Accept. NOTE: This operation is final.

3.3.4.13 Kill All. Kill all active sets.

This is an action item, you'll be asked if it is OK to kill all active sets.

3.3.4.14 Sort. Sort x or y of a set.

Select set to sort. Select which component (x or y) as a key. Select the order of the sort. Press the button marked **Accept** to sort the set in place.

3.3.4.15 Write Set(s). Write a set(s) to disk.

Select the set to write or "All" for all sets. Fill in the format to use to write, syntax is C, default is "%lf %lf". Fill in the item marked "Write to file" with the filename to write. Press the button marked **Accept**. Note: A complete dump of ACE/gr may be accomplished by selecting All sets, All active graphs and toggling Embed parameters. The generated file may be read as a normal data file and will contain the necessary information to completely reconstruct all graphs in the current session.

3.3.4.16 Reverse Order. Exchange points in a set.

Select set to reverse. Press the button marked **Accept** to reverse the order of a set.

3.3.4.17 Coalesce Sets. Merge active sets to an inactive set.

Select set to receive points from all active sets. This selected set should not be an active set. Press the button marked **Accept** to merge all sets to the selected set.

3.3.4.18 Swap. Exchange one set with another.

Select the sets and the graphs these sets reside. Click on **Accept**. The contents of the sets plus all parameters associated with the display of the sets are exchanged.

3.3.4.19 Pack. Clicking on Pack renumbers sets such that there are no gaps in the set structure.

This is an action item, you'll be asked if it is OK to pack the sets.

3.3.5 Region Operations

3.3.5.1 Define Region. Define a region of interest on a graph or graphs.

Select the region, there are 5.

Select the type of region:

Inside polygon - Points inside a closed, non-intersecting polygon are considered inside the region.
Outside polygon - Points outside a closed, non-intersecting polygon are considered inside the region.
Above, below, to the left, and to the right - Regions defining a half-plane, points inside the half- plane are considered in the region.

Select the type of linkage, either a particular graph or all graphs, the default is the current graph.

Press **Accept** to make the pointer ready for the region definition. In the polygonal region type, define the region by successive clicks with the left mouse button, use the right mouse button to register the polygon. In the line case, define the line by clicking on the beginning point and end point of the line.

3.3.5.2 *Evaluate in Region*. Evaluate an expression applied to points within a region.

Select the region to use. Enter the expression, regions are referred to by Rn.x or Rn.y where "n" is the number of the region to use. Functions available are the same as those in "Evaluate expressions" described above.

Examples: R0.X = R0.X - 1

3.3.5.3 *Clear Region*. Remove a defined region.

Select the region to remove and press "Apply".

3.3.5.4 *Extract Points*. Extract points from a region to a set.

Select the region to use. Select the set to receive the points. Select the graph to put the set. Click on **Accept** to combine all points within the specified region to the specified set and graph.

3.3.5.5 *Delete Points*. Delete points in a region.

Select the region to use. Click on **Accept** to delete all points in all sets within the specified region.

3.3.5.6 *Area/Perimeter*. Compute the area and perimeter of a region. This item doesn't belong here as it does not use the region structures. Click on Area or Perimeter and use the mouse to define the region. The area or perimeter will be displayed in the text items Area = or Perimeter = whichever the case may be. Use the right mouse button to close the region.

3.3.6 Edit/Create Sets

Allows loading of a set created using the vi editor.

Opens a vi edit window. Enter the points to go in the new set. Exit the editor with :x or :wq. A new set will be created with the points entered into the file.

3.3.7 Point Operations

Operations in **Point Operations** use the pointer to query sets about points, delete points in a set, and move points. In all cases where interaction is required, pressing the right mouse button with the pointer in the drawing area cancels the operation.

3.3.7.1 *Find Points*. Report on a point in a set.

Click on Find points. Position the pointer close to the data point to identify.

Press the left mouse button. The set, the location in the set, and (X, Y) for the datum nearest to the pointer will be displayed in a popup. Press the right mouse button to shut off the Find point feature.

3.3.7.2 *Tracker*. Track points in a set.

Visit each point of a set in a sequential manner. Click nearest the set to track, and use the left mouse button to go forward through the set and the middle mouse button to go backwards through the set. The right mouse button exits tracking.

3.3.7.3 Delete Points. Deletes points in a set.

Press Delete point with the left mouse button to activate. Position the pointer close to the data point to delete. Press the left mouse button.

3.3.7.4 Add Points. Add points to a set.

Select the set and press Add points to append points to the selected set by clicking on the location where the point is desired. This can be used to (crudely) digitize data from pages pasted on the screen.

3.3.7.5 Move Points. Move point to a new location.

Press Move with the left mouse button to activate. Position the pointer close to the data point to move. Press the left mouse button. Move the pointer to the new location and press the left mouse button again to register the point's new location.

3.3.7.6 Move X Only. Move point X value location.

Click on Move X only to activate. Position the pointer close to the data point to move.

Press the left mouse button. Move the pointer to the new location and press the left mouse button again to register the point's new location.

3.3.7.7 Move Y Only. Move point Y value location.

Click on Move Y only to activate. Position the pointer close to the data point to move. Press the left mouse button. Move the pointer to the new location and press the left mouse button again to register the point's new location.

3.3.7.8 Distance, dy/dx, Angle. Calculates distance, dx, dy or angle of 2 specified points.

Click on Distance, dy/dx, angle to activate. Position the pointer close to the starting point. Press the left mouse button. Click again at another point to form a straight line. The results are the distance, slope, and the angle from horizontal.

3.3.7.9 Goto. Puts the pointer in the specified location.

Enter the X, Y of the point to go to, and press Goto point to have the pointer warp to that position on the drawing area.

3.3.8 Block Data

Create sets from block data.

Select the type of set and the columns to use for each vector of the new set. Press **Accept** to create the set. Note that reading block data does not affect the scaling of the graph so when a set is created using this popup, the set may not appear within the graph's scaling limits - use AS (Autoscale) to set the scale of the graph if needed (or use View/Define world).

3.4 Graph

The selections in the Graph pulldown menu allow the modification of graph parameters. There are popups for setting the scaling of the axes, the number of tick marks to display, legends, writing annotative text, and drawing objects such as lines and boxes.

ACE/gr uses 3 coordinate systems in drawing a graph. These are the world, viewport, and device coordinate systems (following Foley and Van Dam). The world coordinates system is the one in which the data are defined and constitutes the user's coordinate system. The viewport coordinate system (in ACE/gr) is a rectangle defined by the points (0.0, 0.0) or the lower left corner of the device and (1.0, 1.0) or the upper right corner of the device. The pipeline for drawing objects on the screen or hardcopy device is a pair of linear transformations that carry points in world coordinates to the viewport and then on to the device coordinate system. The upshot is that an object located in world coordinates can be drawn in various parts of the screen or hardcopy device depending on the scaling while an object located in viewport coordinates remains in the same spot regardless of the world scaling. Strings, lines, boxes, and the graph legend can be located in either world or viewport coordinates.

3.4.1 Graph Operations

3.4.1.1 Activate. Make a graph eligible for receiving the graph focus and other operations.

Select the graph to activate and press **Accept**.

3.4.1.2 Copy. Copy a graph to another graph.

Select the graph to copy from and the graph to copy to and press **Accept**. Note: Copying a graph will automatically conceal one of the graphs as they will both have the same viewport settings. Some adjustments in the viewport settings of either or both graphs will be required for each graph to be displayed in a non-overlapping manner.

3.4.1.3 Swap. Exchange two graphs.

Select the first graph then the second and press **Accept** to exchange the contents of the two graphs. This can be used to adjust the order in which graphs are drawn, as graphs are drawn in numerical order starting with graph 0.

3.4.1.4 Kill. Make a graph inactive and free all storage associated with sets.

Select the graph to kill and press **Accept**.

3.4.1.5 Focus. Set the graph focus, the focus policy and toggle the display of focus markers.

Select the graph to have the focus, the focus policy, and the toggle for the focus markers and press **Accept**. Note: Most operations in ACE/gr act on the current graph. Not noticing which graph has the current focus can be a source of frustration when working with ACE/gr - if you plan to work extensively with a particular graph, it might be useful to set the focus policy to Set to fix ACE/gr's notion of the current graph.

3.4.1.6 Show. Toggle the display of a one or more graphs.

Select which graph or graphs to show, then press **Accept**. The default is to show all active graphs. This item can relieve some of the drudgery when working with multiple graphs, especially when some of the graphs contain large data sets.

3.4.1.7 Set Graph Type. Set the current graph type to XY, log-linear, linear-log, log-log, bar or stacked bar.

Select the graph type, and press Accept. Note: As of this writing, very little is done to ensure that the data is acceptable for log plots.

3.4.1.8 Arrange Graphs. Place several graphs in a non-overlapping manner.

Select the number of rows and the number of columns, the packing method, the vertical and horizontal spacing between graphs in viewport coordinates, the start of the first graph in viewport coordinates, and how wide and how tall each graph should be. Graphs are laid out in column major order starting from the lower left. So, given 3 columns and 2 rows the graphs will be laid out as follows:

```
1 3 5
0 2 4
```

Given 3 columns and 3 rows:

```
2 5 8
1 4 7
0 3 6
```

Note that graphs are numbered from 0. The packing selection packs the graphs vertically, horizontally or both. Use this item when there are several graphs with the same X or Y axis scaling so graphs on the outside of the packing arrangement provide the tick and axis labeling for all graphs in that row or column. In the 3x3 example above, selecting packing "both", graphs 3, 6 will have X-axis tick labels, and graphs 1, 2 will have Y-axis tick labels. Graphs 4, 5, 7, 8 will have neither X or Y axis tick labels, while graph 0 will have both.

3.4.1.9 Overlay. Overlay one graph onto another. This can be used to plot sets of different scale in what will appear to be the same graph.

3.4.1.10 Invert/Flip Axes. Reverse the direction of the X or Y axes or exchange the X and Y axes.

3.4.1.11 Image. Read and place an X window dump file on the drawing area. At the present time, only one image per session is allowed.

3.4.2 World Scaling

Define the world coordinate system by filling in the items Xmin, Xmax, Ymin, and Ymax. The plot is drawn in world coordinates in a rectangle described by the two points (Xmin, Ymin), (Xmax, Ymax). The input is scanned for arithmetic expressions so setting Xmin = -PI and Xmax = PI is legal input (see the section on transformations for a description of the syntax and available functions). The tick spacing can also be set in this popup by filling in the items for the major and minor spacing for each axis. "Update world/ticks" is used, when, occasionally, the state of the items used to define the world scaling get out of sync with what is actually used to draw the graph, clicking on this item synchronizes the internal values with the displayed values. Press the button marked **Accept** to inform ACE/gr of the changes.

3.4.3 Viewport

Define the viewport by filling in the items Xmin, Xmax, Ymin, Ymax. Viewport coordinates run from (0.0,0.0), the lower left corner of the screen or hardcopy device, to (1.0,1.0), the upper right corner of the screen or hardcopy device. Press the button marked **Accept** to make the change to the new viewport. Press the Pick view button to use the mouse to define the viewport. Take the mouse to the lower left corner of the desired viewport and press the left button (there is no need to hold the button down). A rubberband box will show you the current size of the viewport you may select. When you are satisfied with the view, press the left mouse button again to activate the new viewport. Press the button marked Close to close the Viewport popup.

3.4.4 Autoscale

Use the Autoscale popup to set parameters associated with autoscaling. Select the axis to scale, or none. Selecting none allows the autoscale type and the number of ticks to create to be set but no autoscaling is done. Select the particular set to use or All. The Autoscale type sets the method of autoscaling, either Heckbert or fixed. Heckbert uses a method based on routines posted to the net by Paul Heckbert and generates nice looking tick spacing, but alters the scaling of the axes. The fixed option uses the minimum and maximum values in the set (or over all sets) to set the scale of the axes. Select the number of tick marks to use for defining the tick mark spacing (in the case of Heckbert scaling, these settings are advisory only). The item labeled Apply to: allows the autoscaling of the current graph, or all active graphs.

3.4.5 Draw Options

Set the amount of scrolling, linked scrolling (scrolling the current graph scrolls all graphs), automatic redraw, and whether the screen is erased before refreshing the screen.

3.4.6 Title/Subtitle

The title and subtitle are strings that appear centered at the top of the plot. Fill in the items indicated and press the button marked **Accept**. To change the font, color, or character size used for titles use the items from the popup denoted **Props....**

3.4.7 Ticks/Tick Labels

Set the spacing and type of major and minor tick marks, definition of axes labels, and toggle features associated with the drawing of the graph axes.

31

The top item, **Edit**, selects the axis for which the feature selections will apply. There are two axes defined for each co-ordinate direction, the primary axis and the zero axis. The primary axis will be used the most, the zero axis is, by default, turned off. The button to the right of the Edit item labeled **Props...** allows the axis to be offset by a specified amount in viewport co-ordinates. The item **Axis label** accepts the definition of the label to be used when drawing the axis. Various properties of the label may be set in the popup just below labeled **Axis label props....**

The two items "Major tick spacing" and "Minor tick spacing" set the spacing of major and minor ticks. Tick labels are drawn based on the setting for the major tick spacing. These items are duplicated in the "Define world" popup described previously. Fill in the appropriate blanks and press **Accept** to refresh the display. The units are positive deltas starting with the world minimums. Input is run through the scanner so expressions are allowed. The are special symbols **dx** and **dy** for setting a prescribed number of tick marks. Dx is the distance along the X-axis and dy is the distance along the Y-axis, as defined in the current window of the world coordinate system. Typing at **Major tick spacing** "dx/10" will give 10 major tickmarks. Tick labels, tick marks, the axis bar, may be toggled by the check boxes. Properties for each of these can be set by the Props... popups associated with each item.

Properties for tick labels are:

Font - Which font to use when drawing the labels.
Color - Which color to use.
Line width - Width of line to use when drawing the label.
Char size - Size of characters to use.
Format - Decimal, exponential, power, general plus several time and date formats. To use the time and date formats, the data is required to be in Julian Date format, the long version. Two auxiliary programs, jul2greg and greg2jul, are available with the distributts.
Stagger - Each tick label may be offset with respect to its neighbor, the levels of staggering ranging from 0 to 3. This is especially useful with some of the time and date formats.
Precision - Sets the number of places to display to the right of the decimal point for the labels.
Skip every - Tick labels are drawn at every major tick mark, use the skip factor to circumvent this.
Start labels at - Graph minimum, or specified. To start the labels at some other point than the graph minimum, fill in the text item to the immediate right with the value where tick labels are to begin. There is a similar item in the property sheet for tick marks that can be used to set where the major tick marks begin. between these two, there should be sufficient flexibility.
Stop labels at - Same as above only works at the other end of the axis.
Layout - Allows the tick labels to be drawn horizontally, vertically, or a specified angle (in degrees).
Draw tick labels - Which side of the graph to draw tick labels, either the normal side (the bottom of the graph in the case of the X-axis, or to the left of the graph in the case of the Y-axis), The opposite side, or both. This can be handy when using multiple axes, or overlaying graphs with differing scales.
Sign - Allows the numerical value of the label to be displayed as is, as its absolute value, or ne gated. This can be used to effectively reverse the sense of the axis (negate), or reflect the axis about zero (absolute value).
Apply to - Overrettings to be selected for all axes in the current graph, the current axis the popup.

Properties for tick marks are:

Tick mark direction - In, out, or both. Which direction the ticks point.
Tick marks on - Which side of the graph to draw ticks. The normal side is the bottom of the graph in the case of the X-axis, or to the left of the graph in the case of the Y-axis, the default is to draw ticks on both sides.
Major tick length - Sets the length of major tick marks.

Minor tick length - Sets the length of minor tick marks.

Major grid lines - A check box that when toggled on, grid lines will be drawn at the settings for major tick marks.

Minor grid lines - A check box that when toggled on, grid lines will be drawn at the settings for minor tick marks.

Color, line width, and line style - These items set the characteristics of the lines use to draw ticks and grid lines.

Apply to - Allows the axis select in the Ticks/tick label popup to be overridden, causing the set tings to be selected for all axes in the current graph, the current axis for all graphs, or all axes in all graphs.

Press the button **Accept** when everything is OK, or **Cancel** to close the popup.

Properties for the axis bar are:

Color, line width, and line style - Set the characteristics of the lines use to draw ticks and grid lines.

Press the button **Accept** when everything is OK, or **Cancel** to close the popup.

IMPORTANT NOTE FOR LOG PLOTS: When log scaling is selected for a particular axis or axes, the meaning of major and minor ticks changes slightly. Major ticks should be set to integer values. Use minor ticks to set the number of inter-decade tick marks, these should be set to integer values ranging from 0 (no inter-decade tick marks) to 5. A value of 1 will draw each inter-de cade position, 2 will skip every other position, etc.

3.4.8 Frame

The frame is the box drawn around the clipping region of the graph, and the region inside. Items in the frame popup are:

Frame - Turn the frame on or off.
Style - Either "closed" or "open."
Color - Set the color of the line drawn around the graph.
Line width - Set the width of the line around the graph.
Line style - Set the line style of the line.
Fill graph background - Toggle the filling of the frame.
Background color - The color with which to fill the frame.

Press **Accept** to register the settings, **Close** to close the popup.

3.4.9 Symbols

The items in the symbols popup allow the selection of symbols, lines, and fill style to be used when drawing a set.

Items in the symbols popup are:

Select set - The set to which the following items will apply.

Symbol:

Symbol - Select the symbol to use, or none for no symbol.

Fill - Select the fill option, none, filled, or opaque. The opaque option allows the symbol to be filled with the background color (white as of this writing) overwriting the line used to connect each point.

Size - Set the size of the symbol.

Char - Select a character to use instead of a symbol (not active).

Skip - Select the number of points to skip when drawing symbols.

Legend - A string defining the legend to use for this set.

Apply to - Either this set or all sets.

Line:

Style - Select the line style for the lines connecting the points in the set.

Width - Select the line width for the lines.

Color - The color to use when drawing lines.

Fill:

Fill - Select the type of fill, none, as polygon, to y=0.0, x=0.0, x=graph Xmin, x=graph Xmax, y=graph Ymin, y=graph Ymax.

Fill using - Either Color or Pattern. Color select the color to use, or none.

Pattern - Select the pattern to use or none. There are three additional buttons in this popup that can be used to set the colors, symbols, and line widths in an incremental fashion, avoiding the need to specify a each.

NOTE: There are limits on how large a set can be for the fill to operate correctly. The limit for the display is approximately 8000 points, for the hardcopy drivers, this limit drops to around 800-1000.

Symbols descriptions:

1. No symbol	18. Histogram Y
2. Dot	19. Stair step X
3. Circle	20. Stair step Y
4. Square	21. Bar X
5. Diamond	22. Bar Y
6. Triangle up	23. Range
7. Triangle left	24. Location
8. Triangle down	25. Set #
9. Triangle right	26. Set #, location
10. Plus	27. Bar and whisker (not active as of this writing)
11. X	28. Segments
12. Star	29. Character
13. Impulse at X	30. String (not active as of this writing)
14. Impulse at Y	31. Hi low X
15. Vertical line at X	32. Hi low Y
16. Horizontal line at Y	33. Open/close X
17. Histogram X	34. Open/close Y

NOTE: symbols 16-29 do not appear in the legends.

3.4.10 Error Bars

Error bars are created by reading in the data as a set with error bars (see Files above). Properties of error bars are set by opening the "Error bar props" popup.

Size - Set the length of the error bar.
Line width - Set the line width of the error bar.
Line style - Set the line style of the error bar.
Riser - Toggle the display of the line connecting the error bar with the data point.
Riser line width - Set the width of the riser.
Riser line style - Set the riser line style.
Display - Toggle the display of the error bar.

3.4.11 Legends

Legend items:

Legend on/off - Toggle the display of the graph legend.
Legend location type - Select the coordinate system to use when interpreting the legend x, and legend y items described below. Either the world or viewport coordinates. Legends placed in viewport coordinates make the legends stay put as the plot scale changes. Legends placed in world coordinates will float as the graph scaling is changed.
Legend gap - Specify the vertical gap between legend entries.
Legend length - Specify how long the line representing the set should be.
Legend X - X value of legend location.
Legend Y - Y value of legend location.
Font - Specify the font for the legend labels.
Frame - Toggle a box drawn around the legend. The three items below set the line color, line width, and line style of the box.
Fill frame - Toggle the fill of the box around the legend. The frame fill will obliterate anything beneath, grid lines, sets, etc. Set the type of fill either color or pattern and the corresponding color or pattern.
Legend gap and Legend length - These items refer to the gap between legend items and the length of the legend in units of characters (arbitrary).

Clicking **Place** is a convenient method of placing the legend on the canvas. After pressing this button, move the mouse pointer to the desired location and press again with the left button. The location will be in *world* or *viewport* coordinates as selected above.

Press **Load comments** to make the set comment (generally the file name from where the set originated) the legend label. This affects all active sets in the current graph.

Press the button marked **Accept** to register the legend settings.

Press the button marked **Close** to close the popup.

3.4.12 Strings & Things

The numbers of strings, lines, and boxes are fixed, open File/About to see what the limits are for your version of ACE/gr. To define a text string to be drawn on the canvas:

Select font, pen, justification, rotation (in integer degrees [0,360]), size. Press the button marked "Text" to activate the text writing routines. Move the mouse to the canvas and press the left mouse button at the desired location and type away. Press to advance to the next line - the right mouse button to stop. The text cursor does not behave properly for rotated strings. To redefine string parameters you'll need to press the right button to leave text mode - adjust the parameters and press "Text" again to inform ACE/gr of the changes and resume writing. The cycle "Position in: World | viewport coordinates" allows the string position to be defined in the world coordinate system or viewport coordinates. The difference is that in World coordinates the string position on the display will change as the plot is re-scaled. Placing the string in viewport coordinates will force the string to remain in the same place as the plot scale is changed.

NOTE: It is necessary to press the right button to register the string.

To define a line or box to be drawn on the canvas: The cycle "Position in: World | viewport coordinates" allows the line (box) position to be defined in the world coordinate system or viewport coordinates. The difference is that in World coordinates the line (box) position on the display will change as the plot is re-scaled. Placing the line (box) in viewport coordinates will force the line (box) to remain in the same place as the plot scale is changed.

Lines:
Set the properties of the line using View/Strings & things/Line props if needed. Click on Line. Click at the start of the line, and again at the end of the line to define the line.

Boxes:
Set the properties of the line using View/Strings & things/Line props if needed. Click on Box. Click on one corner of the box, then click again on the opposite corner to dng, press the button marked Move, this should change the cursor to the "move" cursor. Press the left mouse button near the object to be moved. Move the cursor to the new location and press the left mouse button.

Delete:
To delete a line, box, or string, press the button marked Delete, this should change the cursor to a bulls eye cursor. Press the left mouse button near the object to be deleted.

NOTE: The right mouse button cancels all operations (true throughout ACE/gr with respect to mouse operations).

Lines and boxes drawn in world coordinates scale with the plot, to draw fixed length boxes and lines, use viewport coordinates.

3.4.13 Flip X-Y

Release the right mouse button on "Flip X-Y" to swap the positions of X and Y (all sets, tick marks, and labels will be swapped).

3.4.14 Invert X

Reverse the sense of the X-axis making increasing X toward the left. This is performed by making the graph viewport minimum greater than the maximum.

3.4.15 Invert Y

Reverse the sense of the Y-axis making increasing Y toward the left. This is performed by making the graph viewport minimum greater than the maximum.

3.5 Page

Set the dimensions of the drawing area. Presently, there is no connection between the hardcopy drivers and these settings. This means that the printer needs to be set to conform to the dimensions of the drawing area if WYSIWYG is required.

3.5.1 Free

Allow the drawing area to be resized to fit the main ACE/gr window (no scrollbars will appear).

3.5.2 Landscape

Set the drawing area dimensions to landscape, generally this will be larger than the window allocated for ACE/gr, and scroll bars will appear.

3.5.3 Portrait

Set the drawing area dimensions to portrait, generally this will be larger than the window allocated for ACE/gr, and scroll bars will appear.

3.5.4 Fixed

Set the drawing area dimensions to a specified number of pixels, if the drawing area is larger than the ACE/gr window, then scroll bars will appear.

3.5.5 Size...

Open a popup that allows the dimensions of the drawing area to be set to a specified number of pixels in both directions, this only applies when the drawing area orientation is set to Fixed.

3.6 View

Set the display of the status bar, tool bar, locator bar, and properties associated with the display of the pointer position in the Locator bar.

3.6.1 Status Bar

Toggle the visibility of the Status bar that appears at the bottom of the display.

3.6.2 Tool Bar

Toggle the visibility of the Tool bar at the left of the display.

3.6.3 Locator Bar

Toggle the visibility of the Locator bar, at the top of the display.

3.6.4 Set Fixed Point

Select this item, then click on the drawing area to set the position of the fixed point. To set the position more precisely, use the Locator props popup described below. When the fixed point is set, and the display format is (DX, DY), (DIST), or (R, Theta), then give the location relative to the fixed point.

3.6.5 Clear Fixed Point

Set the fixed point to (0.0, 0.0).

3.6.6 Locator Props

Set properties associated with the display of the pointer position in the locator bar.

The choice item at the top of the popup, **Locator: ON/OFF** toggles the continuous update of the pointer as it moves across the drawing area. Locator display type is a choice item the sets the display of the position. There are several options:

[X,Y] The default, display the position in the coordinate system of the current graph.
[DX, DY] Display the delta in X and Y of the position of the mouse relative to the fixed point, or (0, 0) if no fixed point has been selected. The position is given relative to the scaling of the current graph. DX = (CX - FX) and DY = (CY - FY).
[Distance] Display the position of the mouse as the distance from the fixed point, or (0, 0) if no fixed point has been selected. The distance is computed as SQRT((FX - CX)^2 + (FY - CY)^2) where (FX, FY) are the coordinates of the fixed point, and (CX, CY) is the position of the pointer.
[R, Theta] Display the position of the mouse in polar coordinates (R, Theta) relative to the fixed point, or (0, 0) if no fixed point has been selected. R = SQRT((FX - CX)^2 + (FY - CY)^2) and Theta = ATAN2(DY, DX).
[VX, VY] Display the position of the mouse in viewport coordinates. This position is independent of the scaling of the current graph. **[SX, SY]** Display the position of the mouse in screen coordinates. This position is independent of the scaling of the current graph.

The next two categories in the popup set the format and fixed point positions for each coordinate direction and, as they are identical, only setting the format and position of X is described here. The description for Y is identical save that ordinate is replaced by abscissa.

Format X: is a choice item selecting the format of the display of the X position of the pointer. There are several selections here:

Decimal	MM-YY
Exponential	MM-DD
Power (displayed as Decimal)	Month-DD
General	DD-Month
DD-MM-YY	Month (abrev.)
MM-DD-YY	Month

38

Day of week (abrev.)	DD MM' (lon)
Day of week	DD MM'
Day of year	SS.s (lon)
HH:MM:SS.s	MM' SS.s (lon)
MM-DD	Degrees (lat)
HH:MM:SS.s	DD MM' (lat)
MM-DD-YY	DD MM'
HH:MM:SS.s	SS.s (lat)
Degrees (lon)	MM' SS.s (lat)

The Date format requires that the axis be scaled with respect to the long form of the Julian date. The Degrees format is for latitude and longitude. Below the equator latitudes are negative. Longitude is assumed positive east of Greenwich and negative to the west.

Precision X: sets the number of places to display to the right of the decimal point.

Fixed point X: defines the position of the ordinate of the Locator fixed point.

If everything looks OK, click on **Accept** to register the selections, **Reset** to restore the startup values, and Close to close the popup.

4.0 COMMAND INTERPRETER REFERENCE
AND PARAMETER FILE FORMAT

ACE/gr provides a command line interface and the ability to save plot parameters in a file for later retrieval. The syntax for the command line interpreter and parameter files is the same. It is instructive to read a parameter file into the command line interpreter and press the Replay button to verify this feature.

4.1 Reference

Case is ignored by the command line interpreter.

Description of parameters:

color - integer value from 0 to 29.
string - double quote enclosed character string.
integer - any integer >= 0
number - any floating point value >= 0.0
exp. - any expression (includes negative floating point values)
onoff - ON or OFF
torf - TRUE or FALSE
setnum - Set descriptor in the form Sn where n is the number of the set.
graphno - Graph descriptor in the form Gn where n is the number of the graph.
direction - One of UP, DOWN, LEFT, RIGHT, IN, OUT.
opchoice - One of TOP, BOTTOM, LEFT, or RIGHT.
formatchoice - One of:
    DECIMAL - 0.0
    EXPONENTIAL - 0e+-0
    POWER - 1010
    GENERAL - varies between 0.0 and 0e+0
    DDMMYY - requires data in Julian date format
    MMDDYY - requires data in Julian date format
    MMYY - requires data in Julian date format
    MMDD - requires data in Julian date format
    MONTHDAY - requires data in Julian date format
    DAYMONTH - requires data in Julian date format
    MONTHS - requires data in Julian date format
    MONTHL - requires data in Julian date format
    DAYOFWEEKS - requires data in Julian date format
    DAYOFWEEKL - requires data in Julian date format
    DAYOFYEAR - requires data in Julian date format
    HMS - requires data in Degrees format
    MMDDHMS - requires data in Degrees format
    MMDDYYHMS - requires data in Degrees format
    DEGREESLON - requires data in Degrees format
    DEGREESMMLON - requires data in Degrees format
    DEGREESMMSSLON - requires data in Degrees format
    MMSSLON - requires data in Degrees format
    DEGREESLAT - requires data in Degrees format

DEGREESMMLAT - requires data in Degrees format
DEGREESMMSSLAT - requires data in Degrees format
MMSSLAT - requires data in Degrees format

The **Julian date** format is the long form. There are 2 programs in the subdirectory aux/ that can be used to convert data from Gregorian to Julian and vice versa. The **Degrees** format is for latitude and longitude. Below the equator latitudes are negative. Longitude is assumed positive east of Greenwich and negative to the west.

4.2 Exit

Abruptly leave ACE/gr.

4.3 Devices

In the discussion to follow, printer refers to one of:

PSMONOP - PostScript portrait.
PSMONOL - PostScript landscape
MIFP - Maker Interchange Format portrait.
MIFL - Maker Interchange Format landscape.
HPGLP - HPGL portrait.
HPGLL - HPGL landscape.
FILE - print to a file rather than to the device.
HARDCOPY - the current hardcopy device.

PRINT *printer string*
Set the printer command string to string. Example: print psmonop "lpr -Pps".

PRINT TO HARDCOPY
Spool hardcopy output to the printer. This command sets the output destination, but does not generate a hardcopy, see the HARDCOPY command.

PRINT TO FILE
Spool hardcopy output to a file, using the current value of the printer file string (see PRINT TO string above). This command sets the output destination, but does not generate a hardcopy, see the HARDCOPY command.

PRINT TO FILE *string*
Spool hardcopy output to a file named string. This command sets the output destination, but does not generate a file, see the HARDCOPY command. Example: print to file "hardcopy.out".

PRINT TO *printer*
Set the hardcopy device to printer.

HARDCOPY
Print to the current hardcopy device or file.

DEVICE *integer*
Set the screen device to integer. A t the present time, 0 (the X server) is the only value.

4.4 Display

REDRAW
Refresh the display.

AUTO REDRAW *onoff*
Toggle the automatic redrawing of the display.

BACKGROUND COLOR *color*
Set the background color of the drawing area.

CMAP *color, integer, integer, integer*
Set the RGB values of a colormap entry. The value of color ranges from 2 to 29. The three integer values following the color number are values from 0 to 255 and represent red, green, and blue resp. Colors 0 and 1 are black and white (reversed if the -rvideo command line option was selected), and cannot be changed. Xmgr presently dumps core when using this command after star tup, colormap entries can be initialized only. Example: To set colormap entry 5, use

    CMAP 5, 0, 0, 255
    This sets color 5 to solid blue.

PAGE *direction*
Page left, right, up, down, in, out.

PAGE *integer*
Set the amount of scrolling, and integer value giving the amount of scroll in percent of the graph scaling. Example: PAGE 100 would set the amount to scroll left, right, up, down to 100 percent of the graph axis scaling.

PAGE INOUT *integer*
Set the amount to increase or decrease the graph scaling when using the In/Out buttons on the main panel. An integer value expressing the percent to expand or shrink.

LINK PAGE *onoff*
Toggle linked scrolling. Linked scrolling affects the scroll buttons on the main panel, when linked scrolling is on, all graphs are scrolled simultaneously. Linked scrolling OFF makes scrolling affect the current graph only.

4.5 Set Operations

ACTIVATE *setnum number*
Activate a set in the current graph and set the length.

COPY *setnum* **TO** *setnum*
Copy a set to another set.

COPY *graphno.setnum* **TO** *graphno.setnum*
Copy a set from a particular graph to a set in another graph.

Example: copy g0.s0 to g1.s5.

MOVE *setnum* **TO** *setnum*
Move a set to another set.
Example: move s0 to s1.

MOVE *graphno.setnum* **TO** *graphno.setnum*
Move a set from a particular graph to a set in another graph.
Example: move g0.s0 to g1.s5.

KILL *setnum*
Kill a set. Plot parameters are set to their default values.

setnum **POINT** *expr, expr*
graphno.setnum **POINT** *expr, expr*
Add a point to a set, create the set if the set is inactive. The first version adds a point to set setnum in the current graph, the second adds a point to setnum in the graph graphno.

4.6 Block Data

The two block data commands allow block data to be read and sets formed from the active set of block data.

READ BLOCK *string*
Read a file of block data from file string.

BLOCK *xytype string*
Create a set of type *xytype* using columns coded in *string*. *String* describes columns as: "c1:c2:..." Where c1, c2, ... are the numbers of the columns to use in the construction of the set from the block data.
Examples: Create a set with error bars using x from column 1, y from column 2 and the errors in column 5:

    read block "block2.dat"
    block xydy "1:2:5"

Create an XY type data set from columns 7 and 2.
    block xy "7:2"

4.7 Graph Operations

KILL *graphno*
Kill graph graphno.

KILL GRAPHS
Kill all graphs and sets, but not annotative text, lines, and boxes.

FLUSH
Kill all graphs, sets, and annotation.

4.8 Transformations

LOAD VAR *setnum, expr, expr*
Load a set sequentially.

REGRESS *(setnum, number)*
Regress a set where number is the degree of the fit in the range 1-5.

DIFFERENCE *(setnum, number)*
DIFF *(setnum, number)*
Difference a set using the method specified by number.

>   0 - forward difference.
>   1 - backward difference
>   2 - centered difference

INTEGRATE *(setnum)* **INT** *(setnum)*
Integrate a set using a trapezoid rule.

SPLINE *(setnum, start, stop, integer)*
ffttype(*setnum, integer*)
Compute a DFT either forward or inverse, using the DFT or FFT. ffttype is one of:

>   DFT - compute the DFT using the definition.
>   FFT - compute the DFT using an FFT.
>   INVDFT - compute the inverse DFT.
>   INVFFT - compute the inverse using the FFT.

runtype(setnum, number)
Compute a running average, standard deviation, median, maximum, or minimum. runtype is one of the following:

>   RUNAVG - running average.
>   RUNSTD - running standard deviation.
>   RUNMED - running median.
>   RUNMAX - running maximum.
>   RUNMIN - running minimum.

HISTO *(setnum, xmin, xmax, integer)*
Compute a histogram using set setnum. Xmin and xmax are the bounds of the histogram and the integer is how many bins to create in this range.

4.9 Autoscaling

AUTOSCALE
Autoscale the current graph.

AUTOSCALE *setnum*
Autoscale the current graph on a particular set.

AUTOSCALE XAXES
Autoscale the X-axis and the axis at Y = 0 in the current graph.

AUTOSCALE YAXES
Autoscale the Y-axis and the axis at X = 0 in the current graph.

AUTOSCALE *torf*
Suppress autoscaling on startup. Use this command in a data file that uses imbedded parameter settings to set the axes scaling.

4.10 Graph Focus

FOCUS *graphno*
Set the current graph to graphno.

FOCUS SET
Focus fixed on the current graph.

FOCUS *onoff*
Turn the drawing of the focus indicators on or off.

FOCUS FOLLOWS
Focus follows the pointer.

FOCUS CLICK
Set the focus by clicking on a graph.

4.11 Locator

LOCATOR *onoff*
Turn the locator on the front panel on or off.

4.12 IO

SOURCE *sourcetype*
Set the source (disk or pipe) for reading XY data sets.

READ *string*
Read an XY data set.

READ BATCH *string*
Read a batch file.

READ *xytype string*
Read a data set of a particular type.

READ *xytype sourcetype string*
Read a data set of type xytype from source sourcetype.

GETP *string*
Read a parameter file.

PUTP *string*
Write a parameter file.

4.13 Boxes

NOTE: Boxes, used or not, are numbered from 0 to the maximum number of boxes. This is also true of lines and strings.

WITH BOX
Get the next available box and make it current. A box needs to be current before any of the following commands will apply.

WITH BOX *integer*
Get the box numbered integer and make it the current box.

BOX *onoff*
Toggle the display of the current box.

BOX *expr, expr, expr, expr*
Set the location of the box in world or viewport coordinates depending on the value of BOX LOCTYPE. The 4 values represent *xmin, ymin, xmax, ymax* respectively.

BOX LOCTYPE *worldview*
Set the location type of the next box created. World or viewport coordinates.

BOX *graphno*
If the LOCTYPE is WORLD, set the graph to use.

BOX LINESTYLE *integer*
Set the line style to use for the next box created.

BOX LINEWIDTH *integer*
Set the line width of the next box created.

BOX COLOR *integer*
Set the color to use for the box lines.

BOX FILL *filltype*
Set the type of fill to either COLOR or PATTERN.

BOX FILL COLOR *integer*
Set the color to use to fill the next box created.

BOX FILL PATTERN *integer*
Set the pattern to use for the next box created.

BOX DEF
Define the current box using the values set above.

CLEAR BOX
Remove all boxes.

4.14 Lines

ACE/gr uses an array of a data type called lines internally to hold the definition of lines. The actual construction of this data type is not important, but it is important to note that lines, used or not, are numbered from 0 to the maximum number of lines.

WITH LINE
Make the next available line the current line.

WITH LINE *integer*
Make line integer the current line.

LINE *onoff*
Toggle the display of the current line.

LINE *expr, expr, expr, expr*
Set the location of the current line created. The four expressions represent (X1, Y1), (X2, Y2) respectively.

LINE LOCTYPE *worldview*
Set the location type of the current line created. World or viewport coordinates.

LINE *graphno*
If the line LOCTYPE is WORLD, set the graph to use for scaling.

LINE LINEWIDTH *integer*
Set the line width of the current line created.

LINE LINESTYLE *integer*
Set the line style of the current line created.

LINE COLOR *integer*
Set the color of the current line created.

LINE ARROW *integer*
Define a line with or without arrow(s) and where they are located.

    0 - no arrow
    1 - arrow at start of line
    2 - arrow at end of line
    3 - arrow at both ends.

LINE ARROW SIZE *number*
Set the size of the current line's arrowhead.

LINE DEF
Define a line using the current set of line settings.

CLEAR LINE
Remove all lines.

4.15 Strings

ACE/gr uses an array of a data type called plotstr internally to hold the definition of strings. The actual construction of this data type is not important, but it is important to note that strings, used or not, are numbered from 0 to the maximum number of strings.

WITH STRING
Get the next available string.

WITH STRING *integer*
Get the particular string numbered integer.

STRING *onoff*
Toggle the display of the current string.

STRING *expr, expr*
Location of the string, in world or viewport coordinates depending on the value of LOCTYPE.

STRING LOCTYPE *worldview*
Set the location type of the string, either world or viewport coordinates. If the setting is WORLD, then the strings position is affected by changes in the axes scaling, if VIEW, then the string is fixed to that spot in viewport coordinates.

STRING *graphno*
 Set the graph to uses for scaling when the LOCTYPE is WORLD.

STRING LINEWIDTH *integer*
Set the line width of the current string.

STRING COLOR *integer*
 Set the color of the current string.

STRING ROT *integer*
Set the rotation of the current string from-360 to 360 in degrees.

STRING FONT *integer*
 Set the font of the current string.

STRING JUST *integer*
 Set the justification for the current string.

> 0 is left justified
> 1 is right justified
> 2 is centered.

STRING CHAR SIZE *number*
Set the character size of the current string.

STRING DEF *string*
Define the current string using the values set above.

CLEAR STRING
 Remove all strings.

4.16 World

WORLD *xmin, ymin, xmax, ymax*
Set the scaling limits for the current graph.

WORLD XMIN *xmin*
WORLD XMAX *xmax*
WORLD YMIN *ymin*
WORLD YMAX *ymax*
World stack.

PUSH
Push the current graph scaling limits and tick spacing onto the graph's world stack.

POP
Pop the current graph's world stack and set the new scaling limits and tick spacing from the new stack top.

CYCLE
Cycle through the current graph's world stack.

STACK *integer*
 Set the current graph's scaling limits and tick spacing to the value at position integer of the current graph's world stack.

STACK WORLD *expr, expr, expr, expr* **TICK** *expr, expr, expr, expr*
Push specific values onto the current graph's world stack - primarily for use in parameter files generated by ACE/gr.

CLEAR STACK
Clear the current graph's world stack.

4.17 Viewport

VIEW *xmin, ymin, xmax, ymax*
Set the current graph's viewport (where on the device the graph is displayed).

VIEW XMIN *expr*
VIEW XMAX *expr*
VIEW YMIN *expr*
VIEW YMAX *expr*

4.18 Title and Subtitle

TITLE *string*
Set the graph title.

TITLE FONT *integer*
Set the font for the graph title.

TITLE SIZE *number*
Set the character size for the graph title.

TITLE COLOR *integer*
Set the color for the graph title.

SUBTITLE *string*
Set the graph subtitle.

SUBTITLE FONT *integer*
Set the font for the graph subtitle.

SUBTITLE SIZE *number*
Set the character size for the graph subtitle.

SUBTITLE COLOR *integer*
Set the color for the graph subtitle.

4.19 Sets

Setnum, in the following descriptions, refer to the symbolic name of each set, i.e., the letter "s" followed by the integer number of the set. S0 would refer to set 0, s1 to set 1, etc. The following commands for setting set parameters have two prefixes, GRAPHS and SETS, that allow the set ting to be made for all graphs, all sets, or a given set in all graphs. So, to set the line width for set 0 to be 3 in all graphs that have a set 0 active, the command would be:

    graphs s0 linewidth 3

Likewise, to set the line width for all sets in the current graph, give the command:

sets linewidth 3

To set the line width for all sets in all graphs to 3, execute:

graphs sets linewidth 3

setnum **ON**
setnum **IGNORE**
Toggle the active/inactive status of sets. This can be used to force ACE/gr to ignore a set(s), even though data are still attached to the set. For example, assuming S0 is an active set:

S0 ignore

will allow ACE/gr to ignore S0 for all purposes, except any operation that kills a set. Autoscaling will ignore this set, etc - in effect, S0 is dead. To bring it back to life:

S0 on

will reintroduce the set with its data intact. This operation may be performed in either Edit/Set operations/De-activate or in the File/Status popup.

setnum **TYPE** *xytype*
Set the type of set setnum to *xytype*, where xytype is one of xy, xydx, xydy, xydxdx, xydydy, xydxdy, xyz, xyr, or xyhilo.

setnum **FONT** *integer*
Set the font to use when the set type is xyz.

setnum **PREC** *integer*
Set the precision when the set type is xyz.

setnum **FORMAT** *formatchoice*
Set the format to use when the set type is xyz.

setnum **SYMBOL** *integer*
Set the symbol for the set.

setnum **SYMBOL SIZE** *number*
Set the size of the symbol.

setnum **SYMBOL CHAR** *integer*
Set the character to use when using the symbol type character. The value is the decimal representation of the character using the ASCII collating sequence.

setnum **LINESTYLE** *integer*
 Set the line style for the set.

setnum **LINEWIDTH** *integer*
Set the line width for the set.

setnum **COLOR** *integer*
Set the color to use to draw the lines and symbol.

setnum **FILL** *integer*
Set the type of fill for the set.

setnum **FILL WITH COLOR**
Set the type of fill for the set.

setnum **FILL WITH PATTERN**
Set the type of fill for the set.

setnum **FILL COLOR** *integer*
Set the color for the fill if the fill selected is color.

setnum **FILL PATTERN** *integer*
Set the color for the fill if the fill selected is color.

setnum **SKIP** *integer*
Set the number of points to skip before placing a symbol.

setnum **ERRORBAR TYPE** *opchoice*
Set the error bar display type. Opchoice is one of RIGHT, LEFT, or BOTH if the error bar type is xydx or xydxdx, or TOP, BOTTOM, or BOTH if the type is xydy or xydydy. The default in either case is BOTH, i.e. display both error bars.

setnum **ERRORBAR LENGTH** *number*
Set the length of the error bar.

setnum **ERRORBAR LINEWIDTH** *integer*
Set the line width for the error bar.

setnum **ERRORBAR LINESTYLE** *integer*
Set the linestyle for the error bar.

setnum **ERRORBAR RISER** *onoff*
Toggle the display of the error bar riser.

setnum **ERRORBAR RISER LINESTYLE** *integer*
Set the line style for the error bar riser.

setnum **ERRORBAR RISER LINEWIDTH** *integer*
Set the line width for the error bar riser.

setnum **COMMENT** *string*
Set the comment string for the set.

4.20 Legend

LEGEND *onoff*
Toggle display of the legend.

LEGEND LOCTYPE *worldview*
Position the legend in either world or viewport coordinates.

LEGEND VGAP *integer*
Set the vertical gap between legend entries in characters.

LEGEND HGAP *integer*
Specify the gap between the display of the symbol and the legend label in units of characters.

LEGEND LENGTH *integer*
Set the length of the legend in units of characters.

LEGEND *expr, expr*
Set the location of the legend.

LEGEND X1 *expr*
Set the X value of the location.

LEGEND Y1 *expr*
Set the Y value of the legend.

LEGEND FONT *integer*
Set the font to use for the legend labels.

LEGEND CHAR SIZE *number*
Set the size of the characters in the legend label.

LEGEND LINEWIDTH *integer*
Set the line width to use to draw the legend labels.

LEGEND COLOR *integer*
Set the color of the legend labels.

LEGEND BOX *onoff*
Toggle the display of the bounding box for the legend.

LEGEND BOX FILL *onoff*
Toggle the filling of the bounding box for the legend.

LEGEND BOX LINEWIDTH *integer*
Set the line width to use to draw the bound box for the legend.

LEGEND BOX LINESTYLE *integer*
Set the line style to use to draw the bound box for the legend.

LEGEND BOX COLOR *integer*
Set the color of the legend bounding box.

LEGEND BOX FILL COLOR *integer*
Set the color to use for the filled legend bounding box.

LEGEND BOX FILL PATTERN *integer*
Set the pattern to use for the filled legend bounding box.

LEGEND BOX FILL WITH *colpat*
Set the type of fill for the bounding box, either COLOR or PATTERN.

LEGEND STRING *integer*
string Set the legend label for setnum integer.

4.21 Graph Frame

FRAME *onoff*
Toggle the display of the current graph's frame.

FRAME TYPE *number*
Set the type of frame for the current graph:

    0 = rectangle
    1 = Lines along the left and bottom of the graph.

FRAME LINESTYLE *number*
Set the line style of the current graph's frame.

FRAME LINEWIDTH *number*
Set the width of the line for the current graph's frame.

FRAME COLOR *number*
Set the color of the current graph's frame.

FRAME FILL *onoff*
Toggle the fill of the graph frame.

FRAME BACKGROUND COLOR *number*
Set the color to use for filling the current graph's frame.

4.22 Graph Axes

The are two axes in each coordinate direction. In the case of the X coordinate direction, there is one that follows the world scaling, one at Y = 0 and another that may be used to display an alternate scale. The names used by ACE/gr to refer to these axes are, XAXES, ZEROXAXIS, respectively. Likewise along Y there is the YAXIS, ZEROYAXIS. There are names that can be used to refer to all the axes along a coordinate direction or to both directions or to all graphs, these being AXES (both coordinate directions, current graph),

54

XAXES (along X in the current graph), YAXES (along Y in the current graph), and preceding these with the key word GRAPHS, will cause the setting to be made throughout all the active graphs. In the following descriptions, axis refers to the choices described above.

axis onoff
Toggle the display of the axis or axes referred to by axis.

axis **COLOR** *integer*
Set the color for the axis or axes specified by axis.

axis **LINEWIDTH** *integer*
Set the line width for the axis or axes specified by axis.

axis **LINESTYLE** *integer*
Set the line style for the axis or axes specified by axis.

axis **FONT** *integer*
Set the font to use for text for the axis or axes specified by axis.

axis **CHAR SIZE** *number*
Set the character size for text for the axis or axes specified by axis.

4.23 Tick Marks

axis **TICK MAJOR** *onoff*
axis **TICK MINOR** *onoff*
axis **TICK MAJOR** *expr*
Set the tick spacing for major tick marks.

axis **TICK MINOR** *expr*
Set the spacing for minor tick marks.

axis **TICK OFFSETX** *number*
Set the amount to offset the axis, in viewport coordinates in the X direction.

axis **TICK OFFSETY** *number*
Set the amount to offset the axis in viewport coordinates in the Y direction.

axis **TICK ALT** *onoff*
Toggle the use of the alternate map for the axis scaling.

axis **TICK MIN** *number*
Specify the minimum value to use for the alternate map.

axis **TICK MAX** *number*
Specify the maximum value to use for the alternate map.

axis **TICK DEFAULT** *number*
Set the default number of ticks to use when autoscaling.

axis **TICK** *inout*
Set the display of tick marks to IN, OUT, or BOTH.

axis **TICK SIZE** *number*
Set the size of tick marks.

axis **TICK MAJOR SIZE** *number*
Set the size of major tick marks.

axis **TICK MINOR SIZE** *number*
Set the size of minor tick marks.

axis **TICK COLOR** *number*
Set the color to use for tick marks.

axis **TICK MAJOR COLOR** *number*
Set the color to use for major tick marks.

axis **TICK MINOR COLOR** *number*
Set the color to use for minor tick marks.

axis **TICK MAJOR LINEWIDTH** *number*
Set the line width to use for grid lines at major tick marks.

axis **TICK MAJOR LINESTYLE** *number*
Set the line style to use for grid lines at major tick marks.

axis **TICK MINOR LINEWIDTH** *number*
Set the line width to use for grid lines at minor tick marks.

axis **TICK MINOR LINESTYLE** *number*
Set the line width to use for grid lines at minor tick marks.

axis **TICK MAJOR GRID** *onoff*
Toggle the display of grid lines at major tick marks.

axis **TICK MINOR GRID** *onoff*
Toggle the display of grid lines at minor tick marks.

axis **TICK OP** *opchoice*
Set the display of tick marks on the axis to LEFT, RIGHT, or BOTH if the axis is in the X direction, or TOP, BOTTOM, or BOTH if the axis is in the Y direction. The default is BOTH in either case.

axis **TICK TYPE AUTO**
Use the values for major and minor spacing for draw the tick marks. See the next it for tick marks at specified locations.

axis **TICK TYPE SPEC**
Use specified values for drawing tick marks. These values are for major tick marks only.

axis **TICK SPEC** *integer*
Give the number of specified tick marks.

axis **TICK** *number, expr*
Set the value of specified tick mark number integer to value, expr.

4.24 Tick Labels

axis **TICKLABEL PREC** *integer*
Set the number of places to the right of the decimal point when drawing tick labels.

axis **TICKLABEL FORMAT** *format*
Set the format to use for drawing tick labels.

axis **TICKLABEL LAYOUT HORIZONTAL**
Set the angle of the axis tick labels to be horizontal.

axis **TICKLABEL LAYOUT VERTICAL**
Set the angle of the axis tick labels to be vertical.

axis **TICKLABEL SPEC**
Specify the use of the specified tick label angle.

axis **TICKLABEL ANGLE** *number*
Specify the angle to use for drawing the tick labels in degrees from 0 to 360.

axis **TICKLABEL JUST** *justify*
Specify the type of justification to use when drawing the tick label.

axis **TICKLABEL SKIP** *integer*
Set the number of major tick marks to skip before drawing a tick label.

axis **TICKLABEL STAGGER** *integer*
Set the number of characters to use to offset the tick labels.

axis **TICKLABEL OP** *opchoice*
Set the side(s) to draw tick labels, LEFT, RIGHT, or BOTH for tick labels in the Y direction and TOP, BOTTOM, or BOTH for tick labels in the X direction. The default is LEFT for the Y axis tick labels and BOTTOM for the X axis tick labels.

axis **TICKLABEL SIGN** *signchoice*
Set the function to use on the numeric value used to create the tick mark label. One of NORMAL, ABSOLUTE, or NEGATE. The default is NORMAL, i.e., no transformation is applied. ABSOLUTE indicates that the absolute value of the tick label location is used, NEGATE is the negative of the tick mark location. The latter two can be used to achieve a reversal of axis or a reflection effect of the axes. Of course, the data will need to be transformed also.

axis **TICKLABEL START** *expr*
Set the value to use to begin drawing tick labels.

axis **TICKLABEL STOP** *expr*
Set the value to stop drawing tick labels.

axis **TICKLABEL START TYPE SPEC**
Use the specified starting value for drawing tick mark labels.

axis **TICKLABEL STOP TYPE SPEC**
Use the specified stopping value for drawing tick mark labels.

axis **TICKLABEL START TYPE AUTO**
Use the graph minimum to use as the starting point for drawing tick labels, this is the default.

axis **TICKLABEL STOP TYPE AUTO**
Use the graph maximum as the stopping point for drawing tick mark labels. This is the default.

axis **TICKLABEL VGAP** *number*
axis **TICKLABEL HGAP** *number*
axis **TICKLABEL CHAR SIZE** *number*
Set the size of characters to use when drawing the tick mark labels.

axis **TICKLABEL FONT** *integer*
Set the font to use when drawing tick mark labels.

axis **TICKLABEL COLOR** *integer*
Set the color to use for drawing tick mark labels.

axis **TICKLABEL LINEWIDTH** *integer*
Set the line width to use for drawing tick mark labels.

axis **TICKLABEL TYPE AUTO**
Use the values of major tick marks for position the tick mark labels.

axis **TICKLABEL TYPE SPEC**
Specify the tick mark label to use at each major tick mark.

axis **TICKLABEL** *integer, string*
Set the value for the integer'th tick mark label.

4.25 Axes Label Strings

axis **LABEL** *string*
Set the text string to use for the axis label.

axis **LABEL LAYOUT PERP**
Set the layout of the axis label to be perpendicular to the axis.

58

axis **LABEL LAYOUT PARA**
Set the layout of the axis label to be parallel to the axis.

axis **LABEL CHAR SIZE** *number*
Set the character size of the text used for the axis label.

axis **LABEL FONT** *integer*
Set the font to use for the axis label.

axis **LABEL COLOR** *integer*
Set the color to use for the axis label.

axis **LABEL LINEWIDTH** *integer*
Set the line width to use for drawing the axis label.

4.26 Axes Bar

axis **BAR** *onoff*
Toggle the display of the axis bar. The default is OFF.

axis **BAR COLOR** *integer*
Set the color to use for the axis bar.

axis **BAR LINEWIDTH** *integer*
Set the line width to use when drawing the axis bar.

axis **BAR LINESTYLE** *integer*
Set the line style to use for drawing the axis bar.

4.27 Graphs

WITH *graphno*
Set the current graph to graphno.

graphno onoff
Set graphno on or off.

graphno **AUTOSCALE TYPE AUTO**
Set the method of autoscaling to a type developed by Paul Heckbert. This method makes nice tick spacing, but fiddles with scales of the axes.

graphno **AUTOSCALE TYPE SPEC**
Set the method of autoscaling to use the minimum and maximum values of the data.

graphno **HIDDEN** *torf*
Toggle the display of graphno.

graphno **TYPE** *graphtype*
Set the type of graphno to graphtype, where graphtype is one of:

> XY - Linear scaling on both X and Y axes (the default).
> BAR - bar chart.
> STACKEDBAR - stacked bar chart.
> LOGX - logarithmic X axis, linear Y axis.
> LOGY - linear X axis, logarithmic Y axis.
> LOGXY - Log-log graph.

graphno **FIXEDPOINT** *onoff*
Toggle the use of the graph fixed point, i.e., the point used as a reference for the locator display.

graphno **FIXEDPOINT XY** *number, number*
Set the value of the graph fixed point.

graphno **FIXEDPOINT TYPE** *number*
Set the type of display for the locator on the main panel.

graphno **FIXEDPOINT FORMAT** *format format*
Select the format to use for both X and Y in the locator display.

graphno **FIXEDPOINT PREC** *integer, integer*
Set the number of places to display to the right of the decimal point in the locator display for both X and Y.

5.0 XMGR5 EXTENSIONS

Several extensions have been made to the xmgr software developed by Paul Turner, to provide a direct interface to RELAP5 and NRC Databank files, and to provide an easier means of performing calculations using data from these files. This modified version has been designated xmgr5. This document serves as a user manual detailing these extensions.

Modifications to the original xmgr that were made to create xmgr5 include:

- Additions to the Interactive Interface that provide direct access to RELAP5, MELCOR, FRAPCON and NRC Databank files and data.
- Additions to the Command Interpreter (Batch Interface) including:
  - RELAP5 Commands
  - MELCOR Commands
  - FRAPCON Commands
  - SINDA Commands
  - NRC Databank Commands
- An Equation Interpreter that performs calculations using a combination of:
  - Symbolic Variables
  - Operators
  - Math Functions
  - Convenience Functions
  - Steam Table Functions
  - Conversion Functions

Utility Programs were written to convert RELAP5 and MELCOR data files into a compressed, demultiplexed, machine independent binary data format.

Examples of batch files demonstrating the calculational capabilities of xmgr5 are provided at the end of this document.

Questions or comments concerning these extensions should be referred to:

James E. Fisher
e-mail: jef@inel.gov
voice: (208) 526-0571
fax: (208) 526-0528

5.1 Summary of Updates and Bug Fixes

The following changes were made to create version xmgr5.301M:

- A platform independent binary format was implemented for the NRC Databank files. The program will read data files that were created in either the new or old format.

- The user defined variables may now be written out using the WRITEVARS command in either a platform independent binary format or the previous ASCII format. By default, the platform independent binary format is used.

- The status window was modified to include user defined variables. In addition to displaying statistical information, this allows KILL, SAVE, UNSAVE, and PLOT operations to be performed interactively on user defined variables.

- Interactive menus were added under the Files pulldown menu to read and write user defined variable files.

- An UNSAVEVAR command was added to "unsave" variables that had been previously marked with the SAVEVAR command.

The following changes were made to create version xmgr5.301L:

- Interfaces were added to support the MELCOR and FRAPCON codes. MELCOR plot files can either be read directly, or converted into a machine independent demultiplexed format. A utility program, mel2dmx, was written to perform this conversion. The FRAPCON-2 code has been modified to write a machine independent binary plot file for xmgr5.

- An accelerator key, F7, was added to allow direct access to the print button under the "File" pulldown menu.

- All data interfaces were modified to include file names in the comment string associated with each data channel.

- SGI and SUN-OS versions are now supported.

The following changes were made to create version xmgr5.301K:

- The RELAP5 interface was modified to accept a demultiplexed, machine independent, RELAP5 restart plot file. A utility program, r2dmx, was created to convert RELAP5 restart plot files to the new format. In addition to demultiplexing the data to improve access to the data, the new format is machine independent to allow calculations run on various platforms to be analyzed from any other platform. Data compression is also performed to reduce file size.

- Engineering units codes were updated to include additional minor edit variables available in RELAP5 version 3.1

The following changes were made to create version xmgr5.301J:

- The commands WRITEVARS, and READVARS were added to facilitate saving and restoring user defined variables to disk files.

- The DELVAR command was modified to accept the keywords "all," "relap," "nrcdb," "xmgrset" and "user." This allows all variables of a certain type to be deleted with a single command.

- The command SAVEVAR was added. This command is used to mark user defined variables that should not be deleted using the global keywords of DELVAR. SAVEVAR is also used to identify user defined variables that should be written to disk with the WRITEVARS command.

- The output options of the steam table functions were expanded to include constant pressure specific heats: cp, cpf, and cpg; constant volume specific heats: cv, cvf, and cvg; and the ratios of specific heats: gamma, gammaf, and gammag.

The following changes were made to create version xmgr5.301I:

- A Step button was added to the interactive "Commands" menu. This button provides a simpler method of single stepping through a command file.

- Coding was added to detect access to unassigned variables in the equation interpreter.

- Error detection and reporting for the equation interpreter was improved.

- Direct assignment to xmgr data set variables was added.

- Logic was added to improve efficiency by preventing the reloading of RELAP5 and NRC Databank data channels unless the associated file has been changed.

The following changes were made to create version xmgr5.301H:

- Capability to include xmgr data sets as variables in the equation interpreter was added.

- Timestamp information was added to the parameter file.

- Patch level 3 upgrades were incorporated.

The following changes were made to create version xmgr5.301G:

- A convenience function, dropPoints, was added to permit elimination of points from a channel variable based on input from a boolean channel.

- An error in the interpolation algorithm was corrected.

The following changes were made to create version xmgr5.301F:

- An error in parsing NRC databank names in the equation interpreter was corrected.

- A new environmental variable, GR_STEAM, was added to provide a path to the RELAP5 steam table file. Under csh, add "setenv GR_STEAM filename" to your .cshrc file. Where filename contains the path and name of your tpfh2o file.

The following changes were made to create version xmgr5.301E:

- The problems with linking the RELAP5 steam tables on the SUN and HP platforms have been corrected.

- Additional math functions were added:

- stddev - returns a scalar containing the standard deviation of a channel variable.

63

- meanval - returns a scalar containing the mean value of a channel variable.

- minXval, maxXval, minYval and maxYval - return scalars containing the minimum and maximum values of the x and y ranges.

- A minor change was made to the NRC Databank conversion routine. British units of "in H2O" are now converted to SI units of "cm H2O" instead of "Pa".

5.2 Additions to the Interactive Interface

Four menus have been added to the xmgr GUI to allow RELAP5 and NRC database files to be read directly into xmgr data sets.

5.2.1 Read RELAP Data

Located under the Files pulldown menu. This option will open a file selection menu used to open RELAP5 data files. The user must indicate whether the file is a restart plot file, a demultiplexed restart plot file, a strip file written in SI units, or a strip file written in british units. Up to 10 RELAP5 files numbered 0 to 9, may be opened simultaneously. The user may either explicitly specify the file ID number on this menu, or specify that the next available ID be used.

5.2.2 Read MELCOR Data

Located under the Files pulldown menu. This option will open a file selection menu used to open MELCOR data files. The user must indicate whether the file is a MELCOR plot file, or a demultiplexed plot file. Up to 10 MELCOR files numbered 0 to 9, may be opened simultaneously. The user may either explicitly specify the file ID number on this menu, or specify that the next available ID be used.

5.2.3 Read FRAPCON Data

Located under the Files pulldown menu. This option will open a file selection menu used to open FRAPCON data files. Up to 10 FRAPCON files numbered 0 to 9, may be opened simultaneously. The user may either explicitly specify the file ID number on this menu, or specify that the next available ID be used.

5.2.4 Read SINDA Data

Located under the Files pulldown menu. This option will open a file selection menu used to open SINDA data files. Up to 10 SINDA files numbered 0 to 9, may be opened simultaneously. The user may either explicitly specify the file ID number on this menu, or specify that the next available ID be used.

5.2.5 Read Database Data

Located under the Files pulldown menu. This option will open a file selection menu used to open NRC Databank data files. Up to 10 NRC Databank files numbered 0 to 9, may be opened simultaneously. The user may either explicitly specify the file ID number on this menu, or specify that the next available ID be used.

5.2.6 RELAP Data

Located under the Edit pulldown menu. This option will open a menu used to select RELAP5 data channels. The active RELAP5 file may be changed using the "Use File" button. Channels are selected by placing the mouse pointer over the desired channel name and pressing the left mouse button. A filter string may be entered to reduce the number of channels displayed on the menu. The filter string follows the awk language search variable syntax. Examples include:

```
mflowj-10*  display only channels that start with "mflowj-10"
mflowj-[68]*   display only channels that start with "mflowj-" and have a
               6 or 8 as the next character.
```

The "Clear Selections" button clears any currently selected channels. "Clear Current Sets" deletes all sets from the current graph. Either SI or British units may be selected from a radio box. Appropriate conversion factors and y-axis units will be applied to the raw data channels. A pop-up menu allows the user to select the desired time scale units.

5.2.7 MELCOR Data

Located under the Edit pulldown menu. This option will open a menu used to select MELCOR data channels. The active data file may be changed using the "Use File" button. Channels are selected by placing the mouse pointer over the desired channel name and pressing the left mouse button. Menu options are similar to those described above for the RELAP Data menu.

5.2.8 FRAPCON Data

Located under the Edit pulldown menu. This option will open a menu used to select FRAPCON data channels. The active data file may be changed using the "Use File" button. Channels are selected by placing the mouse pointer over the desired channel name and pressing the left mouse button. The user may select between time, power and burnup for the independent variable using the X-Axis Variable button. Additional menu options are similar to those described above for the RELAP Data menu.

5.2.9 SINDA Data

Located under the Edit pulldown menu. This option will open a menu used to select SINDA data channels. The active data file may be changed using the "Use File" button. Channels are selected by placing the mouse pointer over the desired channel name and pressing the left mouse button. The user may select between time, power and burnup for the independent variable using the X-Axis Variable button. Additional menu options are similar to those described above for the RELAP Data menu.

5.2.10 NRC Database Data

Located under the Edit pulldown menu. This option will open a menu used to select NRC Databank data channels. The active NRC Databank file may be changed using the "Use File" button. Channels are selected by placing the mouse pointer over the desired channel name and pressing the left mouse button. A filter string may be entered to reduce the number of channels displayed on the menu. The filter string follows the awk language search variable syntax.

The "Clear Selections" button clears any currently selected channels. "Clear Current Sets" deletes all sets from the current graph. Either SI or British units may be selected from a radio box. Appropriate conversion

factors and y-axis units will be applied to the raw data channels. A pop-up menu allows the user to select the desired time scale units.

An "Interpolate/Decimate/Filter" button is used to access an option menu used to control processing of the data as it is read. The Skip factor is used with the decimate option, while the Begin, End and Step values are used with the interpolate option. A digital filter with up to 30 weighting factors may also be used. These options are further discussed below.

5.3 Additions to the Command Interpreter

The RELAP5 and NRC Databank interfaces can also be accessed though the command interpreter. The additions to the command interpreter that are used for these interfaces are documented in this section. The additions to the command interpreter that are used for the Equation Interpreter are documented in the next section.

5.3.1 RELAP5 Commands

RELAP [fileno] filetype "filename".

Open a RELAP5 data file.

Where:

```
fileno  = Integer value between 0 and 9 identifying the file number.
filetype = restart - RELAP5 restart files
              demux   - Demultiplexed RELAP5 restart files
              strip   - RELAP5 strip files (with SI data)
              stripb  - RELAP5 strip files (with British data)
"filename" = RELAP5 file name.
              Note:  the file name must be in quotes.
```

RREAD [fileno] ["channel"] [units_type]

Specify a RELAP5 data channel and/or the type of units.

Where:

```
fileno = Integer value between 0 and 9 identifying the file number.
"channel" = RELAP5 data channel.  Note, the name must be in quotes.
units_type = si       Convert to SI units (default).
              british Convert to british units.
              done     Process RELAP5 data channels.
```

Specifying the units_type without a RELAP5 channel name will change the default units type. e.g.

```
RREAD british
```

Specifying the units_type following the RELAP5 channel name will change the units type for that channel only. e.g.

```
RREAD "p-100020000" british
```

The key word, done, must follow the list of RELAP5 data channel names. It instructs the software to load the data corresponding to the selected RELAP5 channel names into xmgr data sets. The done keyword may be entered either with the last channel name or on a subsequent RREAD line.

RTIME time_units

Set the units type for the time axis.

Where:

```
time_units = seconds
             minutes
             hours
             days
             fortnights
```

RSIZE [fileno] number

Change the default memory requirements for RELAP5 data.

Where:

```
fileno =     Integer value between 0 and 9 identifying the file number.
number =     maximum number of points to be entered per data channel.
             (default=10000).
```

RELAP5 STRIP are read completely into memory. The memory requirements for STRIP can be determined from:

Total memory requirement = (# of channels on strip file) * RSIZE * (# of bytes/float)

It may be desirable to increase RSIZE to read data sets containing more than 10000 time frames or to decrease RSIZE to increase program efficiency if shorter data sets are used.

5.3.2 MELCOR Commands

MELCOR [fileno] filetype "filename".

Open a MELCOR data file.

Where:

```
fileno      = Integer value between 0 and 9 identifying the file number.
filetype    = PTF - MELCOR plot files
              demux   - Demultiplexed MELCOR plot files
"filename" = MELCOR file name.  Note, the file name must be in quotes.
```

MREAD [fileno] ["channel"] [units_type]

Specify a MELCOR data channel and/or the type of units.

Where:

```
fileno = Integer value between 0 and 9 identifying the file number.
"channel" = MELCOR data channel.  Note, the name must be in quotes.
units_type = si       Convert to SI units (default).
             british Convert to british units.
             done    Process MELCOR data channels.
```

Specifying the units_type without a MELCOR channel name will change the default units type. e.g.

```
MREAD british
```

Specifying the units_type following the MELCOR channel name will change the units type for that channel only. e.g.

```
MREAD "BUR-N-SE_107" british
```

The key word, done, must follow the list of MELCOR data channel names. It instructs the software to load the data corresponding to the selected MELCOR channel names into xmgr data sets. The done keyword may be entered either with the last channel name or on a subsequent MREAD line.

MSIZE [fileno] number

Change the default memory requirements for MELCOR data.

Where:

```
number =      maximum number of points to be entered per data channel.
              (default=10000).
```

5.3.3 FRAPCON Commands

FRAPCON [fileno] "filename".

Open a FRAPCON data file.

Where:

```
fileno = Integer value between 0 and 9 identifying the file number.
"filename" = FRAPCON file name.  Note, the file name must be in quotes.
```

FCREAD [fileno] ["channel"] [units_type]

Specify a FRAPCON data channel and/or the type of units.

Where:

```
fileno = Integer value between 0 and 9 identifying the file number.
"channel" = FRAPCON data channel.  Note, the name must be in quotes.
units_type = si       Convert to SI units (default).
             british Convert to british units.
             done    Process FRAPCON data channels.
```

The key word, done, must follow the list of FRAPCON data channel names. It instructs the software to load the data corresponding to the selected FRAPCON channel names into xmgr data sets. The done keyword may be entered either with the last channel name or on a subsequent FCREAD line.

FCINDP x-axis

Set the independent variable type for FRAPCON data.

Where:

```
x-axis = time   Use time in seconds (default).
         power  Use linear power in kw/m or kw/ft.
         burnup Use burnup in MWD/kg.
```

5.3.4 SINDA Commands

SINDA [fileno] "filename"

Open a SINDA data file.

Where:

```
fileno = Integer value between 0 and 9 identifying the file number.
"filename" = SINDA file name.  Note, the file name must be in quotes.
```

SREAD [fileno] ["channel"] [units_type]

Specify a SINDA data channel and/or the type of units.

Where:

```
fileno = Integer value between 0 and 9 identifying the file number.
 "channel" = SINDA data channel.  Note, the name must be in quotes.
 units_type = si       Convert to SI units (default).
              british  Convert to british units.
              done     Process SINDA data channels.
```

The key word, done, must follow the list of SINDA data channel names. It instructs the software to load the data corresponding to the selected SINDA channel names into xmgr data sets. The done keyword may be entered either with the last channel name or on a subsequent SREAD line.

5.3.5 NRC Databank Commands

NRCDB [fileno] "filename"

Open an NRC Database data file.

Where:

69

```
fileno = Integer value between 0 and 9 identifying the file number.
"filename" =    NRC Database file name. Note, the file name must be in
                quotes.
```

NREAD [fileno] ["channel"] [units_type]

Specify an NRC Database data channel and/or the type of units.

Where:

```
fileno = Integer value between 0 and 9 identifying the file number.
 "channel" =  NRC Database data channel.  Note, the name must be in quotes.
 units_type = si      Convert to SI units (default).
              british Convert to british units.
              done    Process RELAP5 data channels.
```

Specifying the units_type without a data channel name will change the default units type. e.g.

NREAD british

Specifying the units_type following the NRC Databank channel name will change the units type for that channel only. e.g.

```
NREAD "PV-270" british
```

The key word, done, must follow the list of data channel names. It instructs the software to load the data corresponding to the channels into xmgr data sets. The done keyword may be entered either with the last channel name or on a subsequent NREAD line.

INTERPOLATE begin end step

Interpolate an NRC Databank channel to obtain a fixed set of x values on an NREAD. This option overrides a DECIMATE command and remains in effect for all NREAD commands until turned off.

Where:

```
begin =     the start time.  This value should be entered in seconds.
end   =     the end time.  This value should be entered in seconds.
step  =     the size of a time step.  This value should be entered in
            seconds.  A value of 0.0 will turn off this option.
```

DECIMATE n

Skip values an NRC Databank channel to obtain a fixed set of x values on an NREAD. This option overrides an INTERPOLATE command and remains in effect for all NREAD commands, until turned off.

Where:

```
n =  skip factor.  A value of 2 will force future NREAD commands to skip
     every other point.  A value of 0 will turn off this option.
```

70

DIGFLTR onoff

Turn the digital filter on or off. This digital filter is only applied to NRC Databank channels during an NREAD command.

DFWEIGHT n weight

Enter a weighting factor for the digital filter.

Where:

n   =   weight index. Should be an integer value between 0 and n-1 where n is the number of points in the filter. A maximum of 30 weighting factors may be used.

weight = weighting factor. The weighting factor for this position. The sum of all weighting factors should total to 1.0.

The following commands create and activate a digital filter with seven points that will be applied during future NREAD commands.

```
DFWEIGHT    0    0.01881066
DFWEIGHT    1    0.113684
DFWEIGHT    2    0.226243
DFWEIGHT    3    0.282525
DFWEIGHT    4    0.226243
DFWEIGHT    5    0.113684
DFWEIGHT    6    0.01881066
DIGFLTR  ON
```

5.4 Equation Interpreter

Although xmgr provides fairly extensive capability to manipulate data channels, there are two drawbacks associated with using xmgr to perform calculations:

•   Data from external sources must first be read into either an xmgr data set or one of four fixed length scratch arrays. The data is then manipulated by referring to the set number associated with the data. For example, to plot the difference between two RELAP5 data channels, the data is first read into xmgr data sets, then a new set is created containing the results of the calculation. The original two data sets are then eliminated and the result contained in set 2 is plotted. This is illustrated by the following xmgr5 input:

```
RREAD 0 restart "rstplt"
RREAD 0 "p-10102000"
RREAD 0 "p-10103000" done
s2.x = s1.x
s2.y = s0.y - s1.y
KILL s0
KILL s1
REDRAW
```

71

While this approach works well for small calculations, it becomes cumbersome for more complex calculations that may involve well over 30 data channels. The user is limited to a maximum of 30 data sets at any given time, and must keep track of what information is contained in each data set.

- No interpolation is performed when operations are performed between xmgr data sets. This prevents performing operations between sets unless the x-axis values are aligned between the two sets. In many cases, it is desirable to perform calculations using a combination of data that is based on different x-axis values.

An equation interpreter has been added to xmgr5 to resolve these problems, and to permit the user to more easily manipulate data obtained from RELAP5 and the NRC Databank. The features incorporated into this interpreter include:

- Equations can be written using RELAP5 or NRC Databank channel names directly.

- Variables can be created with user defined names that can provide indication of their contents. There is no upper limit set on the number of user defined variables that may be created. Variables can be created to represent either scalar values or channels.

- Interpolation is automatically performed whenever an operator acts on two channels to ensure the operator is applied to y values with common x values. The resultant channel contains the x values from the first argument that are contained within the range of the second argument.

- Several convenience functions are provided to create user defined channels, to combine channels, to extract points from a channel and to insert points into a channel.

- Steam table functions are provided.

Using the equation interpreter, the previous example can then be written as follows:

```
RELAP 0 restart "rstplt"
 CALC "<deltaP_dome> = R0_p-10102000 - R0_p-10103000"
 PLOTVAR "<deltaP_dome>"
 REDRAW
```

CALC "eqn"

Evaluate a user defined equation containing functions, operators and symbolic variables.

Where:

"eqn" is a user defined equation.

Symbolic variables can be either:

- scalars - such as Psat

- RELAP5 channels - such as R0_p-102010000, these are the RELAP5 channel names prefixed by "Rn_" where n corresponds to the RELAP5 file number (0-9).

- MELCOR channels - such as M0_(HS-TEMP_801), these are the MELCOR channel names prefixed by "Mn_(" and ends with a ")" where n corresponds to the MELCOR file number (0-9).

- FRAPCON channels - such as F0_hoop_strain_08, these are the FRAPCON channel names prefixed by "Fn_" where n corresponds to the FRAPCON file number (0-9).

- SINDA channels - such as S0_temp_HC_01, these are the SINDA channel names prefixed by "Sn_" where n corresponds to the SINDA file number (0-9).

- nrcdb channels - such as N0_(TW-123). In this case the NRC Databank channel name is prefixed by "Nn_(" and ends with a ")" where n is the NRC Databank file number (0-9).

- xmgr data sets - the syntax for xmgr data sets is "gm.sn" where m is the graph number and n is the set number. A variable such as g0.s2 will correspond to the data contained in set number 2 of graph number 0.

- vectors or channels - such as <Psat>, must be enclosed with <> typically used to store results of calculations.

Examples include:

```
RELAP 0 restart "relap_rst1"
NRCDB 0 "osu_data"
CALC "<deltaP> = R0_p-102010000 - N0_(P-123)"
CALC "<deltaP> = <deltaP>^2 * 12.0"
CALC "<result> = <deltaP> >= 1.e6"
```

Notes:

1. Everything within the quotes is case sensitive this includes function calls and variable names.

2. A line continuation symbol "\" can be used to create multi-line equations. It must be the last character entered on the line.

3. Whenever an operator acts on two channels, interpolation is automatically performed to ensure the operator is applied to y values with common x values. The resultant channel contains the x values from the first argument that are contained within the range of the second argument.

4. Data may not be assigned to either RELAP5 or NRC Databank channels. For example, expressions such as "R0_p-102010000 = <deltaP>" are not permitted.

PLOTVAR "var"

Load a user defined variable into an xmgr data set. The next available data set of the current graph is used.

Where:

```
"var" is the channel variable to be plotted. Note, var must be in quotes.
```

73

e.g.

```
    PLOTVAR "N0_PV-270"
    PLOTVAR "<deltaP>"
    SAVEVAR  "var"
```

Mark a variable to be saved. This command is used to mark user defined variables that should not be deleted using the global keywords of DELVAR. SAVEVAR is also used to identify user defined variables should be written to disk with the WRITEVARS command.

Where:

```
    "var" is the variable to be saved.  Note, the var must be in quotes.
```

UNSAVEVAR "var"

Un-mark a saved variable. This command is used to mark user defined variables that should not be written to disk with the WRITEVARS command. It has no effect unless the variable has been previously saved with a SAVEVAR command. This command, used with SAVEVAR and WRITEVARS can be used to write variables to separate files.

Where:

```
    "var" is the variable to be unsaved.  Note, the var must be in quotes.
```

DELVAR "var"

Delete a variable. This allows variables to be deleted to free up memory.

Where:

```
    "var" is the variable to be deleted.  Note, the var must be in quotes.
```

The following global keywords may also be used for var:

```
    all     - delete all variables.
    relap   - delete all relap5 variables. (Rn_xx-yyy)
     nrcdb    - delete all NRC Databank variables. (Nn_(xxx)
    xmgrset - delete all xmgr set variables. (Gn.Sm)
    user    - delete all user defined variables. (<xxx>)
```

WRITEVARS [file_type] "filename"

Write all user defined variables marked with the SAVEVAR command to a disk file.

Where:

```
    file_type = PIB   Create a platform independent binary file. (default)
                ASCII Create an ASCII data file.
    "filename"   is the name of the disk file.  Note, the contents of an
                 existing file will be overwritten.
```

74

READVARS "filename"

Read all user defined variables contained on a disk file.

Where:

```
"filename"   is name of the disk file.  Note, any existing variables that
             are also contained in the file will be overwritten.
```

5.4.1 Operators

Table 1 contains a list of the operators supported by the equation interpreter. These include both arithmetic and boolean operators. Boolean operators return either 1.0 for true or 0.0 for a false condition and the results or boolean operations can be used in arithmetic calculations. These operators are similar to those used in the "C" language with the exception that the "^"symbol is used here for exponentiation, rather than as a bit-wise OR operator. The precedence and associativity of the operators is also the same as in the "C" language.

Unless otherwise noted, each of these operators may be applied to any combination of constants, scalar variables and channel variables. Operations involving channel variables act on the dependant variables only, so a string such as "<p_dome> - 100.0" subtracts 100.0 from each y-value in <p_dome>.

If an operation is performed between two channel variables, interpolation will be performed as needed to ensure calculations are performed on the same independent variables. The resulting channel variable will contain the x-values from the left argument that are contained within the right argument. Extrapolation is not performed beyond the range of either channel variable.

Table 1. Precedence and associativity of operators.

Operators	Associativity	Comments
( )	left to right	
-	right to left	Unary Minus
^	right to left	Exponentiation, the power must be a scalar value.
* /	left to right	Multiplication, division.
+ -	left to right	Addition, subtraction.
< <= > >=	left to right	Boolean comparison operators.
== !=	left to right	Boolean equals and not-equals comparison operators.
&&	left to right	Boolean AND
| |	left to right	Boolean OR
=	right to left	Assignment

5.4.2 Math Functions

Table 2 contains a list of the arithmetic functions supported by the equation interpreter. Except as noted, these functions may be applied to any combination of constants, scalar variables and channel variables. As with operators, functions involving channel variables act on the dependant variables only, so a string such as "max(<p_dome>, 100.0)" returns a channel variable containing the x-values of <p_dome> and y-values containing the maximum of either each y-value in <p_dome> or 100.0.

75

If a function is performed between two channel variables, interpolation is performed as needed to ensure calculations are performed on the same independent variables. The resulting channel variable will contain the x-values from the left argument that are contained within the right argument. Extrapolation is not performed beyond the range of either channel variable.

Table 2. Arithmetic functions.

Function	Description
sin(x)	sine of x
cos(x)	cosine
tan(x)	tangent of x
asin(x)	arcsin(x)
acos(x)	arccos(x)
atan(x)	artan(x)
sinh(x)	hyperbolic sine of x
cosh(x)	hyperbolic cosine of x
tanh(x)	hyperbolic tangent of x
exp(x)	exponential function of x
ln(x)	natural logarithm of x, x >0
log(x)	natural logarithm of x, x >0
log10(x)	base 10 logarithm of x, x >0
sqrt(x)	square root of x
ceil(x)	smallest integer not less than x as a double
floor(x)	largest integer not greater than x as a double
abs(x)	absolute value of x
min(x,y)	minimum of x and y
max(x,y)	maximum of x and y
bound(s1, s2, x)	x bounded by scalars s1 and s2, s1<s2
deriv(var)	Return derivative of var as a channel variable.
integrate(var)	Return the integral of var as a channel variable.
meanval(var)	Return the mean value of channel variable var as a double.
stddev(var)	Return the standard deviation of channel variable var as a double.
minXval(var)	Return the minimum value of the x-range of channel variable var as a double.
maxXval(var)	Return the maximum value of the x-range of channel variable var as a double.
minYval(var)	Return the minimum value of the y-range of channel variable var as a double.
maxYval(var)	Return the maximum value of the y-range of channel variable var as a double.

1. Arguments x and y represent. either constants, scalar variables, or channel variables.

5.4.3 Convenience Functions

Table 3 contains a list of convenience functions implemented in the equation interpreter. The shiftx function allow the independent variables to be shifted by a scalar value.

The combine function can be used to change the independent variable. It is used to create a new channel variable that contains x-values based on the y-values of the first argument and y values based on the y-values of the second argument. Interpolation is performed as required to ensure that the independent variables of each argument are properly aligned.

The newSet function creates a new channel variable containing (x2-x1)/dx + 1 points, with the first point being (x1, y1) and the last being (x2, y2). If a zero or negative value of dx is entered, the new channel variable will contain only the (x1, y1) point.

The getYval and putPt functions are useful for manipulating individual data points. getYval will return the y-value of a channel variable associated with a given x-value. putPt can be used to insert individual points into a channel variable.

The integrate and deriv functions are used to calculate the integral and derivative of channel variables respectively.

The functions meanval, stddev, minXval, minYval, maxXval, and maxYval each take a single channel variable as an argument and return a scalar value corresponding to the mean, standard deviation, minimum x-range, minimum y-range, maximum x-range, and maximum y-range, respectively.

Table 3. Convenience functions.

Function	Description
shiftx(dx, var)	Shift channel variable var along x-axis by dx, var is not affected. Returns a channel variable.
combine(var1, var2)	Create a channel variable, using y-values from var1 for x-values and y-values from var2 for y-values.
newSet(x1, x2, dx, y1, y2)	Create a channel variable, with x-values ranging from x1 to x2 incremented by dx, and y-values ranging from y1 to y2.
putPt(x, y, var)	Insert point (x, y) into var, var is not affected. The new, point is inserted in sequence along the x-axis. Returns a channel variable.
getYval(x, var)	Return the y-value in var associated with an x-value of x.
dropPoints(var1,var2)	Return a channel variable containing the points from var2 that do not have corresponding zero values in var1.

1. Arguments var, var1 and var2 represent channel variables. Arguments x, y, x1, x2, dx, y1,and y2 are scalars.

5.4.4 Steam Table Functions

The RELAP5 steam table functions are implemented in this version. These function calls require a RELAP5 water property file, tpfh2o. If this file is not located in your current directory, either add the environmental variable GR_STEAM filename to your shell run command file (.login or .cshrc file for csh, or .bashrc file for bash), or add the command line argument-steam filename when starting xmgr5. Where filename contains the path and file name of the water property file.

Under csh, add:

    setenv GR_STEAM /path/tpfh20

Under bash add:

    setenv GR_STEAM /path/tpfh20
    export GR_STEAM

examples:

    setenv GR_STEAM /usr/local/bin/tpfh2o
    setenv GR_STEAM $HOME/relap/r5m32/run/tpfh20

Each steam table function requires two search variable types and one output variable type. The syntax used for each function is:

out_in1in2(in1var, in2var)

where:

> *in1* is the first search property. The list of supported input properties is given in Table 4.
> *in2* is the second search property. The list of supported input properties is given in Table 4.
> *in1var* is either a constant, a scalar variable, or channel variable containing the value(s) of the first search variable. The values should be in the appropriate SI units indicated in Table 4.
> *in2var* is either a constant, a scalar variable, or channel variable containing the value(s) of the second search variable. The values should be in the appropriate SI units indicated in Table 4.
> *out* is an output property. The list of output properties is provided in Table 5.

For example, to calculate the critical mass flux using the Henry-Fauske model at a constant pressure of 10. MPa and a temperature from channel variable <hot_leg_temp>, storing the result into a new variable <crit_mass_flux>, enter:

CALC "<crit_mass_flux> = ghf_PT(10. e6, <hot_leg_temp>)"

Table 4. Steam table function input combinations.

Second Search Property

		P Pa	T K	v m3/kg	rho kg/m3	u J/kg	h J/kg	x	a
F r i o r p s e t r t y	P	na	X	X	X	X	X	X	X
	T	na	na	X	X	n	n	X	X
	v	na	na	na	n	n	n	n	n
	rho	na	na	na	na	n	n	n	n
	u	na	na	na	na	na	n	n	n
	h	na	na	na	na	na	na	n	n
	x	na	na	na	na	na	na	na	n
	a	na	na	na	na	na	na	na	na

X - implemented.
n - not currently implemented.
na - not applicable.

Table 5. Steam table function outputs.

Symbol	Decription
P	Pressure (Pa)
T	Temperature (K)
v	Specific volume (m^3/kg)
rho	Density (kg/m^3)
u	Internal Energy (J/kg)
h	Enthalpy (J/kg)
s	Entropy (J/kg-K)
cp	Const. Press. Spec. Heat (J/kg-K)
cv	Const. Vol. Spec. Heat (J/kg-K)
gamma	Ratio of Spec. Heats cp/cv
x	Quality
a	Void Fraction
vf	Specific volume - liquid (m^3/kg)
vg	Specific volume - vapor (m^3/kg)
rhof	Density - liquid (kg/m^3)
rhog	Density - vapor (kg/m^3)
uf	Internal Energy - liquid (J/kg)
ug	Internal Energy - vapor (J/kg)
hf	Enthalpy - liquid (J/kg)
hg	Enthalpy - vapor (J/kg)
sf	Entropy - liquid (J/kg-K)
sg	Entropy - vapor (J/kg-K)
cpf	Const. Press. Spec. Heat - liquid (J/kg-K)
cpg	Const. Press. Spec. Heat - vapor (J/kg-K)
cvf	Const. Vol. Spec. Heat - liquid (J/kg-K)
cvg	Const. Vol. Spec. Heat - vapor (J/kg-K)
gammaf	Ratio of Spec. Heats cp/cv - liquid
gammag	Ratio of Spec. Heats cp/cv - vapor
state	Fluid property state
gcrit	Critical mass flux (kg/m^2s)
ghem	Critical mass flux - HEM model (kg/m^2s)
ghf	Critical mass flux - Henry-Fauske model (kg/m^2s)

5.4.5 Conversion Functions

Several units conversion functions have been included to simplify converting data into and out of the standard SI units required by the steam table routines. Table 6 contains a list of these conversion functions. The syntax of each of these functions is:

outvar = convFunc(invar)

where:

outvar is the scalar or channel variable that will contain the result.
invar is the scalar or channel variable that will be converted.
convFunc is the Conversion Function identified in Table 6.

Table 6. Conversion functions.

Property	Conversion Function	Description
Pressure	Pa2psia	Convert Pa to psia.
	psia2Pa	Convert psia to Pa.
Temperature	K2C	Convert deg K to deg C
	K2F	Convert deg K to deg F
	K2R	Convert deg K to deg R
	C2K	Convert deg C to deg K
	C2F	Convert deg C to deg F
	C2R	Convert deg C to deg R
	F2K	Convert deg F to deg K
	F2C	Convert deg F to deg C
	F2R	Convert deg F to deg R
	R2K	Convert deg R to deg K
	R2C	Convert deg R to deg C
	R2F	Convert deg R to deg F
Sp. volume	CMpKG2CFpLBM	Convert m3/kg to ft3/lbm.
	CFpLBM2CMpKG	Convert ft3/lbm to m3/kg.
Int. Energy	JpKG2BTUpLBM	Convert J/kg to BTU/lbm.
	BTUpLBM2JpKG	Convert BTU/lbm to J/kg.
Mass Flux	KGpSM2LBMpSFT	Convert kg/m2 to lbm/ft2.
	LBMpSFT2KGpSM	Convert lbm/ft2 to kg/m2.

For example, to convert a pressure channel, <P_hot_leg_psia>, from psia to Pa, use:

CALC "<P_hot_leg_Pa> = psia2Pa(<P_hot_leg_psia>)"

5.5 Utility Programs

RELAP5 and MELCOR data files may be accessed directly by xmgr5. However, accessing data channels from a large plot files may result in a fairly long delay while the entire data file is read. For this reason, a demultiplexed data format was added to xmgr5 in order to minimize the time required to read data files. Two utility programs, r2dmx and mel2dmx, were created to convert RELAP5 and MELCOR data files into a demultiplexed format, respectively. The demultiplexed data files are written in a machine independent, binary format, allowing calculations run on a wide range of platforms to be analyzed efficiently on another platform.

By default, the demultiplexed data files are written in a compressed format. Decompression is performed on the fly, for selected data channels. The syntax and command line arguments for each program are detailed below.

5.5.1 r2dmx - RELAP5 File Demultiplexer

Syntax: r2dmx [-r rstplt][-d relap.dmx][-mN][-c][-nc][-v][-s][-h]

Options:
        -r : specify RELAP5 restart plot file (default: rstplt)
        -d : specify Demultiplexed file (default: relap.dmx)
        -m : buffer size in MB used to hold relap data 1>N>128 (default: 8)
        -c : compress data (default)   -nc: do not perform compression   -v : verbose mode
        -s : silent mode
        -h : print help message

5.5.2 mel2dmx - MELCOR File Demultiplexer

Syntax: mel2dmx [-p melcor.PTF][-d melcor.dmx][-mN][-c][-nc][-v][-s][-h]

Options:
        -p : specify MELCOR plot file (default: melcor.PTF)
        -d : specify Demultiplexed file (default: melcor.dmx)
        -m : buffer size in MB used to hold MELCOR data 1>N>128 (default: 8)
        -c : compress data (default)
        -nc: do not perform compression
        -v : verbose mode
        -s : silent mode
        -h : print help message

5.6 Examples

Three examples of using the command interpreter are provided below. The batch files may be executed either interactively through the "Commands" option under the "File" pulldown menu or in batch mode by adding "-batch filename" to the command line.

5.6.1 Example One - Steam Table Functions

This example plots the saturation line on a Temperature vs. Entropy diagram. First a channel variable, <Temperature>, containing 1001 points is created, with the x-range varying from 0 to 1000 and the y-range varying from 273.2 K to 647.3 K is created. Two additional channel variables are then created using the <Temperature> channel variable and the steam table function s_Tq, which returns entropy given a temperature and a quality. Channel variable <ent_stm> contains the entropy values associated with a quality of 1.0, while channel variable <ent_liq> contains the entropy values associated with a quality of 0.0.

The combine function creates a new channel variable using the y values of the first and second arguments (aligned by their x values), as the x and y values respectively. The channel variables <line_stm> and <line_liq> topy values in their x-rangee values in their y-ranges.

```
CALC "<Temperature>=newSet(0,1000,1,273.2,647.3)"
CALC "<ent_stm>=s_Tx(<Temperature>,1.0)"
CALC "<ent_liq>=s_Tx(<Temperature>,0.0)"
CALC "<line_stm>=combine(<ent_stm>,<Temperature>)"
CALC "<line_liq>=combine(<ent_liq>,<Temperature>)"
PLOTVAR "<line_stm>"
PLOTVAR "<line_liq>"
xaxis  label "Entropy (J/kg-K)"
yaxis  label "Temperature (K)"
title "Saturation Line" AUTOSCALE
```

5.6.2 Example Two - Plotting RELAP5 vs NRC Databank

This example demonstrates using data from multiple files. Two RELAP5 files and one NRC Databank file are opened. Calculations are then performed using data from each source, and the results are plotted.

```
#
# open data files
#
RELAP 0 DEMUX    "/kkju0/kkj/case1.dmx"
RELAP 1 RESTART "/kkju0/kkj/case2.rstplt"
NRCDB 0 "/home/sbwr_data/dec/sbwl12-FILT.bin"
#
# Calculate the pressure difference for each relap case
# CALC "<deltaP_C1> = R0_p-100010000 - R0_p-240010000"
CALC "<deltaP_C2> = R1_p-100010000 - R1_p-240010000"
#
# convert to psi
#
CALC "<deltaP_C1> = Pa2psia(<deltaP_C1>)"
CALC "<deltaP_C2> = Pa2psia(<deltaP_C2>)"
#
# find the absolute value of the error
#
CALC "<deltaP_Err_C1> = abs(<deltaP_C1> - N0_(DP-120))"
CALC "<deltaP_Err_C2> = abs(<deltaP_C2> - N0_(DP-120))"
#
# calculate the maximum of the two error curves
#
CALC "<max_DP_err> = max(<deltaP_Err_C1>, <deltaP_Err_C2>)"
#
# plot results
#
PLOTVAR "<max_DP_err>"
yaxis  label " Delta P Error (psi)"
AUTOSCALE
PRINT TO FILE "deltPerr.ps"
HARDCOPY
```

5.6.3 Example Three - ROSA Upper Plenum Mass and Energy Balance

This example provides a fairly extensive demonstration of the calculational capabilities available using the equation interpreter.   A mass and energy balance is performed using data obtained from the ROSA test facility.

82

```
NRCDB 0 "/nrclu1/zoo/ap-cl-03/apcl03-FILT.bin"
GETP "rosa.par"
#
#
# MASS AND ENERGY BALANCE ###########
#
#
CALC "UP_h1 = 0.1484"
CALC "UP_h2 = 0.5674"
CALC "UP_h3 = 1.9154"
CALC "UP_h4 = 2.1260"
CALC "UP_vol1 = 0.0204"
CALC "UP_vol2 = 0.0782
CALC "UP_vol3 = 0.3497"
CALC "UP_vol4 = 0.3880"
#
###############################################
# calculate time dependent liquid levels for each subvolume
##################################################################
#
CALC "<UP_liqh1> = bound(0.0,UP_h1,NO_(CL-UP))"
CALC "<UP_liqh2> = bound(UP_h1,UP_h2,NO_(CL-UP))"
CALC "<UP_liqh3> = bound(UP_h2,UP_h3,NO_(CL-UP))"
CALC "<UP_liqh4> = bound(UP_h3,UP_h4,NO_(CL-UP))"
#
PLOTVAR "<UP_liqh1>"
PLOTVAR "<UP_liqh2>"
PLOTVAR "<UP_liqh3>"
PLOTVAR "<UP_liqh4>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "Liquid Level (m)"
AUTOSCALE
PRINT TO FILE "UP_LiqLev.ps"
HARDCOPY
FLUSH
GETP "rosa.par"
#
################################################################
# calculate liquid volumes
################################################################
#
CALC "<UP_liqVol1> = (<UP_liqh1>*UP_vol1)/UP_h1"
CALC "<UP_liqVol2> = ((<UP_liqh2>-UP_h1)*(UP_vol2-UP_vol1))\
                /(UP_h2-UP_h1)"
CALC "<UP_liqVol3> = ((<UP_liqh3>-UP_h2)*(UP_vol3-UP_vol2))\
                /(UP_h3-UP_h2)"
CALC "<UP_liqVol4> = ((<UP_liqh4>-UP_h3)*(UP_vol4-UP_vol3))\
                /(UP_h4-UP_h3)"
#
PLOTVAR "<UP_liqVol1>"
PLOTVAR "<UP_liqVol2>"
PLOTVAR "<UP_liqVol3>"
PLOTVAR "<UP_liqVol4>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "Liquid Volume (m3)"
```

83

```
AUTOSCALE
PRINT TO FILE "UP_LiqVol.ps"
HARDCOPY
FLUSH
GETP "rosa.par"
#
############################################################
# calculate vapor volumes
############################################################
#
CALC "<UP_vapVol1> = UP_vol1-<UP_liqVol1>"
CALC "<UP_vapVol2> = (UP_vol2-UP_vol1-<UP_liqVol2>)"
CALC "<UP_vapVol3> = (UP_vol3-UP_vol2-<UP_liqVol3>)"
CALC "<UP_vapVol4> = (UP_vol4-UP_vol3-<UP_liqVol4>)"
#
PLOTVAR "<UP_vapVol1>"
PLOTVAR "<UP_vapVol2>"
PLOTVAR "<UP_vapVol3>"
PLOTVAR "<UP_vapVol4>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "UP_ Vapor Volume (m3)"
AO FI"
HARDCOPY
FLUSH
GETP "rosa.par"
#
############################################################
# calculate rhof & rhog
############################################################
#
CALC "<UP_rhof1> = rhof_Tx(N0_(TE-E049F-PV), 0.0)"
CALC "<UP_rhof2> = rhof_Tx(N0_(TE-E049F-PV), 0.0)"
CALC "<UP_rhof3> = rhof_Tx(N0_(TE-E060F-PV), 0.0)"
CALC "<UP_rhof4> = rhof_Tx(N0_(TE-E060F-PV), 0.0)"
#
PLOTVAR "<UP_rhof1>"
PLOTVAR "<UP_rhof2>"
PLOTVAR "<UP_rhof3>"
PLOTVAR "<UP_rhof4>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "UP_Liquid Density (kg/m3)"
AUTOSCALE
PRINT TO FILE "UP_rhof.ps"
HARDCOPY
FLUSH
GETP "rosa.par"
#
CALC "<UP_rhog1> = rhog_Tx(N0_(TE-E049F-PV), 1.0)"
CALC "<UP_rhog2> = rhog_Tx(N0_(TE-E049F-PV), 1.0)"
CALC "<UP_rhog3> = rhog_Tx(N0_(TE-E060F-PV), 1.0)"
CALC "<UP_rhog4> = rhog_Tx(N0_(TE-E060F-PV), 1.0)"
#
PLOTVAR "<UP_rhog1>"
PLOTVAR "<UP_rhog2>"
PLOTVAR "<UP_rhog3>"
```

84

```
PLOTVAR "<UP_rhog4>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "UP_Vapor Density (kg/m3)"
AUTOSCALE
PRINT TO FILE "UP_rhog.ps"
HARDCOPY
FLUSH
GETP "rosa.par"
#
############################################################
# calculate liquid and vapor mass in each volume and sum
############################################################
#
CALC "<UP_mf1>  =  <UP_liqVol1>*<UP_rhof1>"
CALC "<UP_mf2>  =  <UP_liqVol2>*<UP_rhof2>"
CALC "<UP_mf3>  =  <UP_liqVol3>*<UP_rhof3>"
CALC "<UP_mf4>  =  <UP_liqVol4>*<UP_rhof4>"
#
PLOTVAR "<UP_mf1>"
PLOTVAR "<UP_mf2>"
PLOTVAR "<UP_mf3>"
PLOTVAR "<UP_mf4>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "Liquid Mass in UP_ Subvolumes (kg)"
AUTOSCALE
PRINT TO FILE "UP_mf.ps"
HARDCOPY
FLUSH
GETP "rosa.par"
#
CALC "<UP_mg1>  =  <UP_vapVol1>*<UP_rhog1>"
CALC "<UP_mg2>  =  <UP_vapVol2>*<UP_rhog2>"
CALC "<UP_mg3>  =  <UP_vapVol3>*<UP_rhog3>"
CALC "<UP_mg4>  =  <UP_vapVol4>*<UP_rhog4>"
# PLOTVAR "<UP_mg1>"
PLOTVAR "<UP_mg2>"
PLOTVAR "<UP"ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "Vapor Mass in UP_ Subvolumes (kg)"
AUTOSCALE
PRINT TO FILE "UP_mg.ps"
HARDCOPY
FLUSH
GETP "rosa.par"
#
############################################################
# calculate total mass in each volume
############################################################
#
CALC "<UP_mass1>  =  <UP_mf1> + <UP_mg1>"
CALC "<UP_mass2>  =  <UP_mf2> + <UP_mg2>"
CALC "<UP_mass3>  =  <UP_mf3> + <UP_mg3>"
CALC "<UP_mass4>  =  <UP_mf4> + <UP_mg4>"
```

85

```
#
PLOTVAR "<UP_mass1>"
PLOTVAR "<UP_mass2>"
PLOTVAR "<UP_mass3>"
PLOTVAR "<UP_mass4>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "Mass in UP_ Subvolumes (kg)"
AUTOSCALE
PRINT TO FILE "UP_masses.ps"
HARDCOPY
FLUSH
GETP "rosa.par"
#
##########################################################
#
calculate total UP mass
##########################################################
#
CALC "<UP_TotMass>=<UP_mass1>+<UP_mass2>+\
                   <UP_mass3>+<UP_mass4>"
#
PLOTVAR "<UP_TotMass>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "Total UP Mass (kg)"
AUTOSCALE
PRINT TO FILE "UP_TotMass.ps"
HARDCOPY FLUSH
#
##########################################################
# Normalize Total Mass to IC
##########################################################
#
CALC "<clip> = newSet(-20,0,10,1.0,1.0)"
CALC "<UP_TotMass_clip> = <UP_TotMass> * <clip>"
CALC "UP_TotMass_IC = meanval(<UP_TotMass_clip>)"
CALC "<UP_TotMass_Norm> = <UP_TotMass> / UP_TotMass_IC"
#
PLOTVAR "<UP_TotMass_Norm>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "Normalized UP_ Mass"
AUTOSCALE
PRINT TO FILE "NormUP_Mass.ps"
HARDCOPY
FLUSH
#
##########################################################
# calculate Tsat(P)
##########################################################
#
CALC "<UP_Tsat> = T_Px(N0_(PE280A-PV)*1.e6, 1.0)"
#
# define band along Tsat for saturated approximation ########
```

```
#
CALC "<UP_TL> = <UP_Tsat> - 5.00"
CALC "<UP_TU> = <UP_Tsat> + 5.00"
#
# plot Tsat, TU, TL
#
PLOTVAR "<UP_Tsat>"
PLOTVAR "<UP_TL>"
PLOTVAR "<UP_TU>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "Tsat,TU, and TL (K)"
AUTOSCALE
PRINT TO FILE "UP_Tbounds.ps"
HARDCOPY
FLUSH
GETP "rosa.par"
#
############################################################
# Determine if local Temp is within sat. band #############
# sat# = 1 if in band; 0 if not in band
############################################################
#
CALC "<UP_sat1> = (NO_(TE-E049F-PV) > <UP_TL>) && \
                  (NO_(TE-E049F-PV) < <UP_TU>)"
CALC "<UP_sat2> = (NO_(TE-E049F-PV) > <UP_TL>) && \
                  (NO_(TE-E049F-PV) < <UP_TU>)"
CALC "<UP_sat3> = (NO_(TE-E060F-PV) > <UP_TL>) && \
                  (NO_(TE-E060F-PV) < <UP_TU>)"
CALC "<UP_sat4> = (NO_(TE-E060F-PV) > <UP_TL>) && \
                  (NO_(TE-E060F-PV) < <UP_TU>)"
#
CALC "<UP_nosat1> = (NO_(TE-E049F-PV) < <UP_TL>) || \
                    (NO_(TE-E049F-PV) > <UP_TU>)"
CALC "<UP_nosat2> = (NO_(TE-E049F-PV) < <UP_TL>) || \
                    (NO_(TE-E049F-PV) > <UP_TU>)"
CALC "<UP_nosat3> = (NO_(TE-E060F-PV) < <UP_TL>) || \
                    (NO_(TE-E060F-PV) > <UP_TU>)"
CALC "<UP_nosat4> = (NO_(TE-E060F-PV) < <UP_TL>) || \
                    (NO_(TE-E060F-PV) > <UP_TU>)"
#
PLOTVAR "<UP_sat1>"
PLOTVAR "<UP_nosat1>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
xaxis  label "Time (s)"
yaxis  label "Boolean"
AUTOSCALE
PRINT TO FILE "UP_satno1.ps"
HARDCOPY
FLUSH
GETP "rosa.par"
#
PLOTVAR "<UP_sat2>"
PLOTVAR "<UP_nosat2>"
title "ROSA AP-CL-03"
world xmin -500
world xmax 7500
```