GRIDAL, A PACKAGE TO DRAW BACKGROUNDS FOR X/Y PLOTS

David J. Kennison
NCAR, P.O. Box 3000, Boulder, Colorado 80307-3000
email: kennison@ncar.ucar.edu (128.117.64.4)

GRIDAL, A PACKAGE TO DRAW BACKGROUNDS FOR X/Y PLOTS

David J. Kennison
NCAR, P.O. Box 3000, Boulder, Colorado 80307-3000
email: kennison@ncar.ucar.edu (128.117.64.4)

1.0 INTRODUCTION

The package named GRIDAL allows one to draw backgrounds for X/Y plots. Included are routines for drawing grids, perimeters, and pairs of axes. Internal parameters of GRIDAL control such things as the color indices and line width scale factors used to draw various parts of the background, the formats to be used for labels, the sizes of the characters to be used for labels, the amount by which labels are to be offset from the axes, and the routine to be called to draw the labels (the SPPS routine WTSTR or the PLOTCHAR routine PLCHHQ).

1.1 Usage

Each of the routines GRID, GRIDL, HALFAX, PERIM, PERIML, and GRIDAL draws a background of some sort within the part of the plotter frame occupied by the current GKS viewport, as follows:

- GRID draws an unlabelled grid.
- GRIDL draws a labelled grid.
- HALFAX draws a pair of intersecting axes.
- PERIM draws an unlabelled perimeter.
- PERIML draws a labelled perimeter.
- GRIDAL draws any of the above.

The positioning of ticks and grid lines is determined partly by the arguments of these routines and partly by the nature of the current mapping from the “user” coordinate system to the “fractional” coordinate system, as determined by the current definitions of the GKS viewport and window and by the values of two internal parameters of the package SPPS: 'LS' and 'MI'. Numeric labels reflect values in the “user” coordinate system.

Note: Calling the SPPS routine SET defines the GKS viewport and window and the values of the parameters 'LS' and 'MI'. The parameter 'LS' determines whether the mappings of user coordinates to fractional coordinates are linear or logarithmic and the parameter 'MI' determines whether the mappings are mirror-imaged or not. For more information about all of this, see the write-up of SPPS and/or the document “The Use of X/Y Coordinates in NCAR Graphics”.

Each of the internal parameters of GRIDAL has a three-character mnemonic name of the form 'xxx'. Each of the routines GAGETC, GAGETI, GAGETR, GASETC, GASETI, and GASETR is intended either to retrieve the value of a specific internal parameter of GRIDAL or to give a new value to it, as follows:

- GAGETC is called with the name of an internal parameter of type CHARACTER and a character variable in which the value of the parameter is to be returned.
- GAGETI is called with the name of an internal parameter of type INTEGER or REAL and an integer variable in which the integer equivalent of the value of the parameter is to be returned.
- GAGETR is called with the name of an internal parameter of type INTEGER or REAL and a real variable in which the real equivalent of the value of the parameter is to be returned.
- GASETC is called with the name of an internal parameter of type CHARACTER and a character expression that is to become the new value of the parameter.
- GASETI is called with the name of an internal parameter of type INTEGER or REAL and an integer expression that is to become the new value of the parameter.
- GASETR is called with the name of an internal parameter of type INTEGER or REAL and a real expression that is to become the new value of the parameter.

These six routines allow one to access all of the internal parameters of GRIDAL in a manner consistent with other NCAR Graphics utilities and are intended to replace some older routines, each of which allows access to a limited subset of the internal parameters. However, the older routines are still available and will remain so for the foreseeable future:

- GACOLR sets the value of internal parameters that control the color of various parts of the background.
- LABMOD sets the value of internal parameters that affect the appearance of labels.
- TICKS and TICK4 set the value of internal parameters that determine the length and direction of ticks.

See the sections “ROUTINES” and “INTERNAL PARAMETERS” for more information.

1.2 Common Arguments

Several of the routines in GRIDAL have arguments MJRX, MNRX, MJRY, and MNRY, which specify the number of major and minor divisions of the horizontal (X) and vertical (Y) axes of the current viewport. The meanings of these parameters depend on the current setting of the internal parameter 'LS' of SPPS:

- If the value of 'LS' implies that the axis is linear: MJRX(Y) specifies the number of major divisions of the X(Y) axis and MNRX(Y) specifies the number of minor divisions within each major division. In each case, the value specifies the number of spaces between grid lines or ticks rather than the number of lines or ticks. Including the ones at the ends of the axes, there is always one more major division line or mark than the number of major divisions specified by MJRX(Y). Similarly, there is always one less minor division line or tick per major division than the number of minor divisions per major division specified by MNRX(Y).
- If the value of 'LS' implies that the axis is logarithmic: Each major division point occurs at a value 10**MJRX(Y) times the previous point. For example, if the minimum X-axis value were 3., the maximum X-axis value 3000. and MJRX 1, then the major division points would be 3., 30., 300., and 3000. If MNRX(Y).LE.10, there are nine minor divisions within each major division. For example, between 3. and 30., there would be minor division points at 6., 9., 12., . . . 27. If MNRX(Y).GT.10., minor divisions are omitted.

2.0 ROUTINES

2.1 GACOLR (KAXS,KLBL,KMJT,KMNT)

Sets the values of internal parameters determining the color of various parts of the background.

2.1.1 Usage

Use the statement

```
CALL GACOLR (KAXS,KLBL,KMJT,KMNT)
```

This statement resets the values of the following internal parameters:

Internal parameter	Value given to it
'CAX'	KAXS
'CLB'	KLBL
'CMJ'	KMJT
'CMN'	KMNT

See the section “INTERNAL PARAMETERS” for more information.

2.1.2 Arguments

KAXS, **KLBL**, **KMJT**, and **KMNT** (input expressions of type INTEGER) are new values for the internal parameters 'CAX', 'CLB', 'CMJ', and 'CMN', specifying the color indices to be used for the axes, the labels, the major ticks/grid lines, and the minor ticks/grid lines, respectively. Values less than zero imply that no call is to be done to set the color before drawing items of the associated type. The default value of each of the four internal parameters is -1.

2.2 GAGETx (PNAM,xVAL)

Retrieves the value of a specified internal parameter.

2.2.1 Usage

Use one of the statements

```
CALL GAGETC (PNAM,CVAL)
CALL GAGETI (PNAM,IVAL)
CALL GAGETR (PNAM,RVAL)
```

to retrieve, in the variable CVAL, IVAL, or RVAL, the value of the parameter specified by PNAM.

2.2.2 Arguments

PNAM (an input expression of type CHARACTER) is a string three or more characters in length, the first three characters of which constitute the name of the internal parameter whose value is to be retrieved.

xVAL (an output variable of type CHARACTER, INTEGER, or REAL, depending on whether “x” is a “C”, an “I”, or an “R”) is the name of a variable into which the value of the internal parameter specified by PNAM is to be retrieved.

- If “x” is a “C”, then the internal parameter must be of type CHARACTER.
- If “x” is an “I” or an “R”, then the internal parameter may be of type INTEGER or REAL; if the type of “xVAL” and the type of the internal parameter do not match, the effect will be the same as that of a mixed-mode arithmetic replacement statement in FORTRAN.

For more information, see the section “INTERNAL PARAMETERS”.

2.3 GASETx (PNAM,xVAL)

Gives a new value to a specified internal parameter.

2.3.1 Usage

Use one of the statements

```
CALL GASETC (PNAM,CVAL)
CALL GASETI (PNAM,IVAL)
CALL GASETR (PNAM,RVAL)
```

to set the value of the parameter specified by PNAM to the value specified by the argument CVAL, IVAL, or RVAL.

2.3.2 Arguments

PNAM (an input expression of type CHARACTER) is a string three or more characters in length, the first three characters of which constitute the name of the internal parameter whose value is to be set.

xVAL (an input expression of type CHARACTER, INTEGER, or REAL, depending on whether “x” is a “C”, an “I”, or an “R”) is the new value to be given to the internal parameter specified by PNAM.

- If “x” is a “C”, then the internal parameter must be of type CHARACTER.
- If “x” is an “I” or an “R”, then the internal parameter may be of type INTEGER or REAL; if the type of “xVAL” and the type of the internal parameter do not match, the effect will be the same as that of a mixed-mode arithmetic replacement statement in FORTRAN.

For more information, see the section “INTERNAL PARAMETERS”.

2.4 GRID (MJRX,MNRX,MJRY,MNRY)

Draws an unlabelled grid.

2.4.1 Usage

Use the statement

```
CALL GRID (MJRX,MNRX,MJRY,MNRY)
```

which is equivalent to the statement

```
CALL GRIDAL (MJRX,MNRX,MJRY,MNRY,0,0,0,0.,0.)
```

2.4.2 Arguments

MJRX, **MNRX**, **MJRY**, and **MNRY** (input expressions of type INTEGER) specify the major and minor divisions of the two axes, as described in the introduction.

2.5 GRIDAL (MJRX,MNRX,MJRY,MNRY,IXLB,IYLB, . . .)

(The remaining arguments are **IGPH**, **XINT**, and **YINT**.)

Draws any of the supported types of backgrounds. Each of the other background-drawing routines is implemented by a call to GRIDAL.

2.5.1 Usage

Use the statement

```
CALL GRIDAL (MJRX,MNRX,MJRY,MNRY,IXLB,IYLB,IGPH,XINT,YINT)
```

2.5.2 Arguments

MJRX, **MNRX**, **MJRY**, and **MNRY** (input expressions of type INTEGER) specify the major and minor divisions of the two axes, as described in the introduction.

IXLB (an input expression of type INTEGER) is defined as follows:

- IXLB = -1 implies that no X axis is to be drawn.
- IXLB = 0 implies that the X axis is to be drawn unlabelled.
- IXLB = 1 implies that the X axis is to be drawn and labelled.

IYLB (an input expression of type INTEGER) is defined as follows:

- IYLB = -1 implies that no Y axis is to be drawn.
- IYLB = 0 implies that the Y axis is to be drawn unlabelled.
- IYLB = 1 implies that the Y axis is to be drawn and labelled.

IGPH (an input expression of type INTEGER) specifies the background type, as follows:

IGPH	X axis	Y axis
0	grid	grid
1	grid	perim
2	grid	halfax
4	perim	grid
5	perim	perim
6	perim	halfax
8	halfax	grid
9	halfax	perim
10	halfax	halfax

XINT and **YINT** (input expressions of type REAL), if IGPH has the value 10, are the user coordinates of the point of intersection of the two axes. For other values of IGPH for which one of the axes is of type HALFAX, XINT and/or YINT specify the position of that axis.

2.6 GRIDL (MJRX,MNRX,MJRY,MNRY)

To draw a labelled grid. Each major division is labelled with its numeric value.

2.6.1 Usage

Use the statement

```
CALL GRIDL (MJRX,MNRX,MJRY,MNRY)
```

which is equivalent to the statement

```
CALL GRIDAL (MJRX,MNRX,MJRY,MNRY,1,1,0,0.,0.)
```

2.6.2 Arguments

MJRX, **MNRX**, **MJRY**, and **MNRY** (input expressions of type INTEGER) specify the major and minor divisions of the two axes, as described in the introduction.

2.7 HALFAX (MJRX,MNRX,MJRY,MNRY, . . .)

(The remaining arguments are **XINT**,**YINT**, **IXLB**, and **IYLB**.)

Draws orthogonal axes intersecting at a specified point and with a specified set of labels.

2.7.1 Usage

Use the statement

```
CALL HALFAX (MJRX,MNRX,MJRY,MNRY,XINT,YINT,IXLB,IYLB)
```

This is equivalent to the statement

```
CALL GRIDAL (MJRX,MNRX,MJRY,MNRY,IXLB,IYLB,10,XINT,YINT)
```

2.7.2 Arguments

MJRX, **MNRX**, **MJRY**, and **MNRY** (input expressions of type INTEGER) specify the major and minor divisions of the two axes, as described in the introduction.

XINT, **YINT**, **IXLB**, and **IYLB** are defined as for GRIDAL, above.

2.8 LABMOD (FMTX,FMTY,NUMX,NUMY,ISZX,ISZY, . . .)

(The remaining arguments are **IXDC**, **IYDC**, and **IXOR**.)

Presets internal parameters controlling the appearance of labels drawn by GRIDAL, GRIDL, . . . et al. LABMOD itself does no plotting and, in order to have any effect, must be called prior to the background-drawing routines for which it is presetting parameters.

2.8.1 Usage

Use a statement like

```
CALL LABMOD (FMTX,FMTY,NUMX,NUMY,ISZX,ISZY,IXDC,IYDC,IXOR)
```

This statement resets the values of the following internal parameters:

Internal parameter	Value given to it
'XLF'	FMTX
'YLF'	FMTY
'XLL'	NUMX
'YLL'	NUMY
'XLS'	REAL(ISZX)
'YLS'	REAL(ISZY)
'XLO'	REAL(IYDC) - not REAL(IXDC)
'YLO'	REAL(IXDC) - not REAL(IYDC)
'XOR'	IXOR

See the section “INTERNAL PARAMETERS” for more information.

2.8.2 Arguments

FMTX and **FMTY** (input expressions of type CHARACTER) contain format specifications for the X-axis and Y-axis numerical labels produced by GRIDAL, GRIDL, HALFAX, or PERIML. The specification must begin with a left parenthesis and end with a right parenthesis and must not be more than ten characters long. Conversions of types E, F, G, and I are allowed; for example, one might use FMTX='(F8.2)' and FMTY='(E10.0)'. The default for both formats is '(E10.3)'.

NOTE: I formats are allowed by this version of GRIDAL; they were not allowed by previous versions.

NUMX (an input expression of type INTEGER) , if non-zero, is the number of characters in each X-axis numeric label; if LBLX is a string produced by the format FMTX, then the label will be the substring LBLX(1:NUMX). If NUMX is 0, then the label will be the substring LBLX(m:n), where LBLX(m:m) is the first non-blank character in LBLX, and LBLX(n:n) is the last non-blank character following LBLX(m:m). Using a non-zero NUMX causes the labels to be centered differently than if a zero value is used. The default value for NUMX is 0.

NUMY (an input expression of type INTEGER) is defined just like NUMX, but applies to Y-axis numeric labels.

ISZX and **ISZY** (input expressions of type INTEGER) are character sizes for the labels, specified in plotter address units, just as for the SPPS routines PWRIT and WTSTR. The default value for both is 10.

IXDC (an input expression of type INTEGER) is the decrement, in plotter address units (PAUs - by default, the plotter frame is 1023 PAUs in width and height), from the left edge of the current viewport to the nearest X address of the label specified by FMTY, NUMY, and ISZY. For example, if the horizontal extent of the current viewport is defined by the normalized device coordinates .1 and .9, and if IXDC is 60, and if there has been no call to the SPPS routine SETI (which can change the size of a PAU), then labels on the Y axis will end at plotter coordinate 43 (.1*1023+1-60). Negative values may be used to put labels on the other side of the viewport; in the example given, changing IXDC to -878 (-.8*1023 -60) would put the labels on the right side of the viewport, with their left edges 60 plotter-coordinate units away from the edge of the viewport. There are two special values of IXDC:

- If IXDC=0, the Y-axis labels will end 20 plotter address units to the left of the viewport (equivalent to using IXDC=20).
- If IXDC=1, Y-axis labels will begin 20 plotter address units to the right of the viewport (equivalent to using IXDC=-20-w, where w is the width of the viewport, in plotter address units).

The default value of IXDC is 20.

When HALFAX is called or when GRIDAL is called with IGPH = 2, 6, or 10, IXDC is the distance from the Y axis, rather than from the minimum viewport coordinate, and the special values 0 and 1 are equivalent to 20 and -20.

IYDC (an input expression of type INTEGER) is the decrement, in plotter address units (PAUs - by default, the plotter frame is 1023 PAUs in width and height), from the bottom edge of the current viewport to the nearest Y address of the label specified by FMTX, NUMX, and ISZX. Note that negative values may be used to put labels above the viewport. There are two special values of IYDC:

- If IYDC=0, the top of the X-axis labels will be 20 plotter address units below the bottom edge of the viewport (equivalent to using IYDC=20).
- If IYDC=1, the bottom of the X-axis labels will be 20 plotter address units above the top edge of the viewport (equivalent to using IYDC=-20-h, where h is the height of the viewport, in plotter address units).

The default value of IYDC is 20.

When HALFAX is called or when GRIDAL is called with IGPH = 8, 9, or 10, IYDC is the distance from the X axis, rather than from the minimum viewport coordinate, and the special values 0 and 1 are equivalent to 20 and -20.

IXOR (an input expression of type INTEGER) specifies the orientation of the X-axis labels:

- IXOR = 0 implies the use of horizontal labels.
- IXOR = 1 implies the use of vertical labels.

The default orientation is horizontal.

2.9 PERIM (MJRX,MNRX,MJRY,MNRY)

Draws an unlabelled perimeter with inward-pointing tick marks. The directions and lengths of tick marks may be changed by calling TICKS and/or TICK4 (which see, below).

2.9.1 Usage

Use the statement

```
CALL PERIM (MJRX,MNRX,MJRY,MNRY)
```

which is equivalent to the statement

```
CALL GRIDAL (MJRX,MNRX,MJRY,MNRY,0,0,5,0.,0.)
```

2.9.2 Arguments

MJRX, **MNRX**, **MJRY**, and **MNRY** (input expressions of type INTEGER) specify the major and minor divisions of the two axes, as described in the introduction.

2.10 PERIML (MJRX,MNRX,MJRY,MNRY)

Draws a labelled perimeter with inward-pointing tick marks. The directions and lengths of tick marks may be changed by calling TICKS and/or TICK4 (which see, below).

2.10.1 Usage

Use the statement

```
CALL PERIML (MJRX,MNRX,MJRY,MNRY)
```

which is equivalent to the statement

```
CALL GRIDAL (MJRX,MNRX,MJRY,MNRY,1,1,5,0.,0.)
```

2.10.2 Arguments

MJRX, **MNRX**, **MJRY**, and **MNRY** (input expressions of type INTEGER) specify the major and minor divisions of the two axes, as described in the introduction.

2.11 TICKS (LMJR,LMNR)

Allows for program control of tick mark length and direction. This routine has been superseded by TICK4, which should be used instead. The statement

```
CALL TICKS (LMJR,LMNR)
```

is equivalent to the statement

```
CALL TICK4 (LMJR,LMNR,LMJR,LMNR)
```

See the description of TICK4, below.

2.12 TICK4 (LMJX,LMNX,LMJY,LMNY)

Allows for program control of tick mark length and direction.

2.12.1 Usage

Use the statement

```
CALL TICK4 (LMJX,LMNX,LMJY,LMNY)
```

This statement resets the values of the following internal parameters:

Internal parameter	Value given to it
'XMJ'	REAL(LMJX)
'XMN'	REAL(LMNX)
'YMJ'	REAL(LMJY)
'YMN'	REAL(LMNY)

See the section “INTERNAL PARAMETERS” for more information.

2.12.2 Arguments

All arguments are in plotter address units (PAUs). By default, one PAU is 1/1023rd of the width of the plotter frame.

Note: A call to the SPPS routine SETI will change the definition of a PAU. Such calls are no longer recommended, but the possibility still exists. See the write-up of SPPS for more information.

LMJX and **LMNX** (input expressions of type INTEGER) are the lengths, in plotter address units, of major and minor ticks on the X axis. The default values are 12 and 8.

LMJY and **LMNY** (input expressions of type INTEGER) are the lengths, in plotter address units, of major and minor ticks on the Y axis. The default values are 12 and 8.

By default, tick marks point inward. Negative values of LMJX, LMNX, LMJY, and LMNY may be used to create outward-pointing tick marks.

3.0 INTERNAL PARAMETERS

GRIDAL has twenty-two internal parameters, each of which affects, in some way, the behavior of the routines that draw backgrounds. The current value of a parameter may be retrieved by calling GAGETC, GAGETI, and/or GAGETR and it may be reset by calling

GASETC, GASETI, and/or GASETR; certain parameters may also be reset by calling other, older, routines (GACOLR, LABMOD, TICKS, and TICK4). The following table lists all of the internal parameters of GRIDAL:

Name	Type	Description
'CAX'	I	The polyline color index to be used for drawing the axes. A negative value implies that no change is to be made in the polyline color index before drawing the axes. The default value of 'CAX' is -1; its value may be changed by calling GASETI, GASETR, or GACOLR (argument KAXS) .
'CLB'	I	The polyline/text color index to be used for drawing labels. (Both are affected because the labels may be drawn using calls to WTSTR or calls to PLCHHQ.) A negative value implies that no change is to be made in the polyline and text color indices before drawing the labels. The default value of 'CLB' is -1; its value may be changed by calling GASETI, GASETR, or GACOLR (argument KLBL).
'CMJ'	I	The polyline color index to be used for major ticks and/or grid lines. A negative value implies that no change is to be made in the polyline color index before drawing major ticks/grid lines. The default value of 'CMJ' is -1; its value may be changed by calling GASETI, GASETR, or GACOLR (argument KMJT).
'CMN'	I	The polyline color index to be used for minor ticks and/or grid lines. A negative value implies that no change is to be made in the polyline color index before drawing minor ticks/grid lines. The default value of 'CMN' is -1; its value may be changed by calling GASETI, GASETR, or GACOLR (argument KMNT).
'LTY'	I	The labelling type. The value 0 implies that the SPPS routine WTSTR is to be called to draw labels, the value 1 that the PLOTCHAR routine PLCHHQ is to be called. The default value of 'LTY' is 0; its value may be changed by calling GASETI or GASETR.
'WAX'	R	The line width to be used for drawing the axes. A value less than or equal to zero implies that no change is to be made in the line width scale factor before drawing the axes. The default value of 'WAX' is 0; its value may be changed by calling GASETI or GASETR.
'WLB'	R	The line width to be used for drawing labels. (Labels drawn using calls to the PLOTCHAR routine PLCHHQ are affected by this; those drawn using calls to the SPPS routine WTSTR are not.) A value less than or equal to zero implies that no change is to be made in the line width scale factor before drawing the labels. The default value of 'WLB' is 0; its value may be changed by calling GASETI or GASETR.
'WMJ'	R	The line width to be used for major ticks and/or grid lines. A value less than or equal to zero implies that no change is to be made in the line width scale factor before drawing. The default value of 'WMJ' is 0; its value may be changed by calling GASETI or GASETR.
'WMN'	R	The line width to be used for minor ticks and/or grid lines. A value less than or equal to zero implies that no change is to be made in the line width scale factor before drawing. The default value of 'WMN' is 0; its value may be changed by calling GASETI or GASETR.

Name	Type	Description
'XLF'	C	The format to be used for labels on the X axis. The character string must begin with a left parenthesis and end with a right parenthesis and it must not exceed ten characters in length. Conversions of types E, F, G, and I are allowed. The default value of 'XLF' is '(E10.3)'; its value may be changed by calling GASETC or LABMOD (argument FMTX).
'XLL'	I	The length of each X-axis label. If 'XLL' is given a non-zero value “n” and LBLX is a string produced by the format 'XLF', then the label will be the substring LBLX(1:n). If, on the other hand, 'XLL' = 0, then the label will be the substring LBLX(m:n), where LBLX(m:m) is the first non-blank character in LBLX and LBLX(n:n) is the last non-blank character following LBLX(m:m). Using a non-zero value for 'XLL' causes labels to be centered differently than if a zero value is used. The default value of 'XLL' is 0; its value may be changed by calling GASETI, GASETR, or LABMOD (argument NUMX).
'XLO'	R	The (vertical) distance of an X-axis label from the X axis. Values between 0 and 1 are interpreted as fractions of the width of the plotter frame, while values greater than 1 are interpreted as plotter address units (PAUs). The value is interpreted as the distance from the bottom edge of the viewport to the nearest Y address of an X-axis label; negative values may be used to put labels above the viewport. The value 0 is treated as equivalent to 20 and the value 1 is treated as equivalent to -20-h, where h is the height of the viewport in plotter address units. The default value of 'XLO' is 20, which means 20 PAUs; its value may be changed by calling GASETI, GASETR, or LABMOD (argument IYDC). (Note that the LABMOD argument IXDC corresponds to 'YLO' and that the LABMOD argument IYDC corresponds to 'XLO', which is perhaps somewhat counterintuitive.)
'XLS'	R	The size (width) of characters in X-axis labels. Values between 0 and 1 are interpreted as fractions of the width of the plotter frame, while values greater than or equal to 1 are interpreted as plotter address units (PAUs). The integral values 0., 1., 2., and 3. mean 8, 12, 16, and 24 PAUs, respectively. Values less than zero are treated as zero. The default value of 'XLS' is 10, which means 10 PAUs; its value may be changed by calling GASETI, GASETR, or LABMOD (argument ISZX).
'XMJ'	R	The length of major ticks on the X axis. ABS('XMJ') specifies the length of major ticks on the X axis; values between 0 and 1 are interpreted as fractions of the width of the plotter frame, while values greater than or equal to 1 are interpreted as plotter address units (PAUs). If 'XMJ' is positive, the ticks point inward; if 'XMJ' is negative, the ticks point outward. The default value of 'XMJ' is 12, which means12 PAUs; its value may be changed by calling GASETI, GASETR, or TICKS (argument LMJX).
'XMN'	R	The length of minor ticks on the X axis. ABS('XMN') specifies the length of minor ticks on the X axis; values between 0 and 1 are interpreted as fractions of the width of the plotter frame, while values greater than or equal to 1 are interpreted as plotter address units (PAUs). If 'XMN' is positive, the ticks point inward; if 'XMN' is negative, the ticks point outward. The default value of 'XMN' is 8, which means 8 PAUs; its value may be changed by calling GASETI, GASETR, or TICKS (argument LMNX).

Name	Type	Description
XOR'	I	The orientation of X-axis labels. The value 0 implies the use of horizontal labels, while the value 1 implies the use of vertical labels. The default value is 0; its value may be changed by calling GASETI, GASETR, or LABMOD (argument IXOR).
'YLF'	C	The format to be used for labels on the Y axis. The character string must begin with a left parenthesis and end with a right parenthesis and it must not exceed ten characters in length. Conversions of types E, F, G, and I are allowed. The default value of 'YLF' is '(E10.3)'; its value may be changed by calling GASETC or LABMOD (argument FMTY).
'YLL'	I	The length of each Y-axis label. If 'YLL' is given a non-zero value “n” and LBLY is a string produced by the format 'YLF', then the label will be the substring LBLY(1:n). If, on the other hand, 'YLL' = 0, then the label will be the substring LBLY(m:n), where LBLY(m:m) is the first non-blank character in LBLY and LBLY(n:n) is the last non-blank character following LBLY(m:m). Using a non-zero value for 'YLL' causes labels to be centered differently than if a zero value is used. The default value of 'YLL'is 0; its value may be changed by calling GASETI, GASETR, or LABMOD (argument NUMY).
'YLO'	R	The (horizontal) offset distance of a Y-axis label from the Y axis. Values between 0 and 1 are interpreted as fractions of the width of the plotter frame, while values greater than 1 are interpreted as plotter address units (PAUs). The value is interpreted as the distance from the left edge of the viewport to the nearest X address of a Y-axis label; negative values may be used to put labels to the right of the viewport. The value 0 is treated as equivalent to 20 and the value 1 is treated as equivalent to -20-w, where w is the width of the viewport in plotter address units. The default value of 'YLO' is 20, which means 20 PAUs; its value may be changed by calling GASETI, GASETR, or LABMOD (argument IXDC). (Note that the LABMOD argument IXDC corresponds to 'YLO' and that the LABMOD argument IYDC corresponds to 'XLO', which is perhaps somewhat counterintuitive.)
'YLS'	R	The size (width) of characters in Y-axis labels. Values between 0 and 1 are interpreted as fractions of the width of the plotter frame, while values greater than or equal to 1 are interpreted as plotter address units (PAUs). The integral values 0., 1., 2., and 3. mean 8, 12, 16, and 24 PAUs, respectively. Values less than zero are treated as zero. The default value of 'YLS' is 10, which means 10 PAUs; its value may be changed by calling GASETI, GASETR, or LABMOD (argument ISZY).
'YMJ'	R	The length of major ticks on theY axis. ABS('YMJ') specifies the length of major ticks on theY axis; values between 0 and 1 are interpreted as fractions of the width of the plotter frame, while values greater than or equal to 1 are interpreted as plotter address units (PAUs). If 'YMJ' is positive, the ticks point inward; if 'YMJ' is negative, the ticks point outward. The default value of 'YMJ' is 12, which means12 PAUs; its value may be changed by calling GASETI, GASETR, or TICKS (argument LMJY).

Name	Type	Description
'YMN'	R	The length of minor ticks on the Y axis. ABS('YMN') specifies the length of minor ticks on theY axis; values between 0 and 1 are interpreted as fractions of the width of the plotter frame, while values greater than or equal to 1 are interpreted as plotter address units (PAUs). If 'YMN' is positive, the ticks point inward; if 'YMN' is negative, the ticks point outward. The default value of 'YMN' is 8, which means 8 PAUs; its value may be changed by calling GASETI, GASETR, or TICKS (argument LMNY).

4.0 ERROR HANDLING

When a GRIDAL routine detects an error condition, it calls the routine SETER, which is the principal routine in the error-handling package for NCAR Graphics. (There is a programmer document describing SETER and associated routines; see that document for complete information about error handling in NCAR Graphics.)

By default, SETER prints a line and STOPs. The line printed will look something like this:

```
ERROR    3 IN GRIDAL - ERROR EXIT FROM GQPLCI
```

The error number (“3”, in the example) may be of use to a consultant (to determine exactly where the error occurred), but is not otherwise meaningful. The actual error message consists of the name of the routine in which the error occurred (“GRIDAL”, in the example), a blank, a minus sign, another blank, and, lastly, a short description of the error.

All errors are “recoverable” in the sense that, if the user program puts SETER into “recovery mode”, control will be returned to the caller of the GRIDAL routine in which the error occurred. In some cases, it is then possible to take remedial action to get around whatever problem has occurred; in any case, the error flag can be cleared and execution of the user’s program can continue.

When SETER is in recovery mode (and, occasionally, even when it is not), error messages may have a somewhat more complicated form, like this:

```
HALFAX/GRIDAL - ERROR EXIT FROM GQPLCI
```

What this particular error message says is that HALFAX called GRIDAL, which detected an error condition (an error exit from a GKS query routine) and called SETER. Upon getting control back from GRIDAL, HALFAX detected the fact that GRIDAL had logged an error. It augmented the error message by prepending its own name, followed by a slash, and then passed control back to the user. Of course, there can be more than two such levels of routine calls indicated in the error message: in a few cases, seven or eight routine names may be listed, each separated from the next by a slash.

The various error conditions in GRIDAL are described in the list below. Each bulleted item includes an error message and a thumb-nail description of the error. The items in the list are arranged in alphabetical order. If you get an error message with one or more prefixed subroutine names, as described above, omit them and look up the result in this list. Note that, since GRIDAL routines sometimes call other routines, elsewhere in NCAR Graphics, that can detect error conditions and call SETER, the error message you get by calling a GRIDAL routine may not be listed here, but in the programmer document for some other package.

- **GACOLR - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time GACOLR was called, there was an unrecovered outstanding error. In this case, GACOLR cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **GAGETC - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time GAGETC was called, there was an unrecovered outstanding error. In this case, GAGETC cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **GAGETC - UNRECOGNIZED PARAMETER NAME**

  The first argument in a call to GAGETC is not one of the legal internal parameter names of GRIDAL.

- **GAGETI - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time GAGETI was called, there was an unrecovered outstanding error. In this case, GAGETI cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **GAGETR - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time GAGETR was called, there was an unrecovered outstanding error. In this case, GAGETR cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **GAGETR - UNRECOGNIZED PARAMETER NAME**

  The first argument in a call to GAGETR is not one of the legal internal parameter names of GRIDAL.

- **GASETC - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time GASETC was called, there was an unrecovered outstanding error. In this case, GASETC cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **GASETC - UNRECOGNIZED PARAMETER NAME**

  The first argument in a call to GASETC is not one of the legal internal parameter names of GRIDAL.

- **GASETI - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time GASETI was called, there was an unrecovered outstanding error. In this case, GASETI cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **GASETR - UNCLEARED PRIOR ERROR**

This error message indicates that, at the time GASETR was called, there was an unrecovered outstanding error. In this case, GASETR cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **GASETR - UNRECOGNIZED PARAMETER NAME**

  The first argument in a call to GASETR is not one of the legal internal parameter names of GRIDAL.

- **GRID - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time GRID was called, there was an unrecovered outstanding error. In this case, GRID cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **GRIDAL - ERROR EXIT FROM GQCLIP**

  The GKS routine GQCLIP, which is called to get the current value of the clipping flag, has returned a non-zero error code.

- **GRIDAL - ERROR EXIT FROM GQLWSC**

  The GKS routine GQLWSC, which is called to get the current value of the line width scale factor, has returned a non-zero error code.

- **GRIDAL - ERROR EXIT FROM GQPLCI**

  The GKS routine GQPLCI, which is called to get the current value of the polyline color index, has returned a non-zero error code.

- **GRIDAL - ERROR EXIT FROM GQTXCI**

  The GKS routine GQTXCI, which is called to get the current value of the text color index, has returned a non-zero error code.

- **GRIDAL - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time GRIDAL was called, there was an unrecovered outstanding error. In this case, GRIDAL cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **GRIDL - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time GRIDL was called, there was an unrecovered outstanding error. In this case, GRIDL cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **HALFAX - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time HALFAX was called, there was an unrecovered outstanding error. In this case, HALFAX cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **LABMOD - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time LABMOD was called, there was an unrecovered outstanding error. In this case, LABMOD cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **PERIM - UNCLEARED PRIOR ERROR**

This error message indicates that, at the time PERIM was called, there was an unrecovered outstanding error. In this case, PERIM cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **PERIML - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time PERIML was called, there was an unrecovered outstanding error. In this case, PERIML cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **TICK4 - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time TICK4 was called, there was an unrecovered outstanding error. In this case, TICK4 cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

- **TICKS - UNCLEARED PRIOR ERROR**

  This error message indicates that, at the time TICKS was called, there was an unrecovered outstanding error. In this case, TICKS cannot continue; it forces the error message for the outstanding error to be printed and then substitutes this one for it.

5.0 EXAMPLES

On a Unix system on which NCAR Graphics has been installed, the command "ncargex" may be used to obtain and run an example showing how to use GRIDAL. The name of the example is "tgrida", so the command one uses is

```
ncargex tgrida
```

To look at the resulting metafile, use the command

```
ctrans tgrida.ncgm
```

To acquire the code for the example without running it, use the command

```
ncargex -n tgrida
```

After modifying the source file, it can be compiled and run using the commands

```
ncargf77 tgrida.f
a.out
```

Then, to look at the resulting metafile, use the command

```
ctrans gmeta
```

(For "ctrans" to work properly, one or more environment variables may need to be set.)

The unmodified example produces nine frames, one frame apiece for each of the routines GRID, GRIDL, PERIM, PERIML, HALFAX, TICK4, and LABMOD and two frames for the routine GRIDAL.

6.0 MISCELLANY

GRIDAL has been a part of the NCAR System Plot Package for many years. As part of the GKS effort, the routines comprising GRIDAL were incorporated into a separate package. The present code is the result of a total re-write in October, 1986.

User-level entry points in the package GRIDAL are as follows: GALBEX, GRID, GRIDAL, GRIDL, HALFAX, LABMOD, PERIM, PERIML, TICKS, and TICK4.

GRIDAL uses the following common block names: GAREIN and GACHAR.

GRIDAL uses the routine SETER and various routines from the package SPPS.

GRIDAL is written in portable FORTRAN-77.