Astronomical Data Analysis Software and Systems IV
ASP Conference Series, Vol. 77, 1995
R. A. Shaw, H. E. Payne, and J. J. E. Hayes, eds.

WIP – An Interactive Graphics Software Package

J. A. Morgan

Astronomy Program, University of Maryland, College Park, MD 20742

Abstract. WIP[1] is an interactive package with a simple to use interface designed to produce high quality graphical output. WIP was developed as part of the Berkeley-Illinois-Maryland Association[2] (BIMA) project and is available via anonymous ftp. Details are presented about the WIP package along with a few examples.

1. Introduction

Astronomers need to generate high quality graphics. There are, in general, two reasons why existing graphics software is limited. First, the software often used to reduce the data can also generate graphics but not of publication quality. Either it generates graphics at a low to moderate quality or it produces figures that contain ancillary information that is not desirable in the final draft. Second, astronomers collaborating with astronomers at other institutions find it helpful to have a common plotting environment at each institution. However, commercial graphics products can not be freely passed from one institution to another making it impossible for non-licensed sites to be able to contribute to the graphics directly.

WIP was written to address these concerns. WIP was developed to be used along with BIMA's Miriad[3] data reduction package (Sault et al. 1995) and is based, at the lowest level, on Tim Pearson's PGPLOT subroutine library.

Because WIP is public domain software and can easily be installed on various machines (and architectures), it provides a common interface to high quality graphics that can be readily available to every researcher. The WIP distribution is available on anonymous ftp[4]. Included in the WIP distribution is a complete manual[5] which contains several sample figures with associated commands, a frequently asked questions section, and a descriptive listing of all available commands.

---

[1]http://bima.astro.umd.edu/bima/wip/wip.html

[2]http://bima.astro.umd.edu/bima/home.html

[3]http://bima.astro.umd.edu/bima/miriad/miriad.html

[4]ftp://ftp.astro.umd.edu/progs/morgan/

[5]http://bima.astro.umd.edu/bima/wip/manual/wip.html

2. The WIP Package

WIP has a very simple user interface which easily provides high quality graphics output. However, in addition to the simplicity of the command line interface, there are four characteristics that make WIP quite powerful. The first of these is probably most important to astronomical researchers: WIP has the ability to read two dimensional images and can sense the image type (Miriad, FITS, IRAF, etc.) automatically. In addition, most image types contain headers that WIP can use when displaying the image as a halftone or contour plot. This greatly simplifies one of the most difficult plotting tasks astronomers face: overlaying of images with different resolutions and spatial extents.

Next, WIP has variables that may be assigned to arbitrarily difficult expressions. These variables allow the user to write command files in a generic way so that they evaluate expressions internally rather than work with hard coded numerical values. Also, the default variable list can be expanded by dynamically creating or freeing additional variables as needed.

WIP also supports user definable macros. Macros can be defined and edited at run time or defined in, and read from, external files. These files can also be automatically loaded each time WIP is started providing a way to expand easily the number of available commands. Additionally, macros (along with the ability to pass arguments) provide a way to write, in a compact way, a collection of repetitive commands and also provide another way to simplify the process of writing, debugging, and enhancing plot files.

Finally, plotting scripts may be controlled internally by a conditional command and a looping construct. The *if* command permits conditional execution of commands. The *loop* command simplifies the commands needed to execute a macro or command multiple times. Both of these constructs (along with the user variables) permit complex graphics to be written quite easily.

2.1. Examples

There are many other features available within the WIP package. Rather than describe or list each feature, some will be illustrated in the examples that follow. Space limitations make it impossible to explain each example in depth; the interested reader will find more details in the manual distributed with the package.

Figure 1 illustrates three of the four major characteristics described above: user variables, macros, and looping. The use of these traits minimizes the number of commands that would otherwise be necessary to generate the figure.

Commands needed to generate Figure 1

```
define flippanel              # $1=Panel #; $2/3=Nx/Ny.
set \0 ($1 % $2) + 1          # Index in the x-direction.
set \1 ($1 \ $2) + 1          # Index in the y-direction.
set \0 \0 + ($2 * ($3 - \1)) # Flip the y index direction.
panel -$2 -$3 \0              # Set the panel.
end

define dosymbol               # $1=Counter; $2=Nx; $3=Ny.
set $1 $1 + 1                 # Increment the Counter.
```

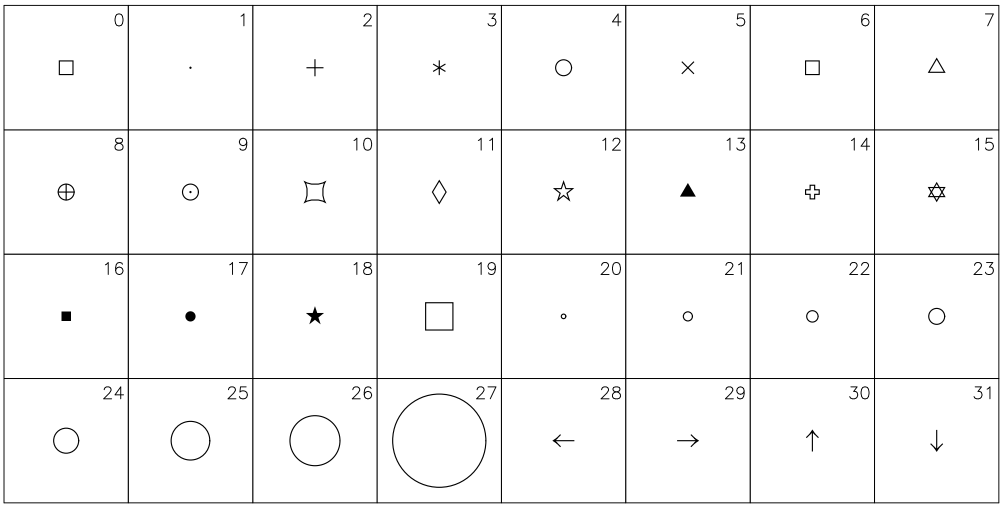

Figure 1. A plot of the standard symbol markers used by WIP.

```
flippanel $1 $2 $3            # Set up the panel.
box bc bc                     # Draw the frame for this panel.
expand 0.5                    # Set a small character size.
mtext T -1.1 0.97 1.0 \[$1]
expand 1.0                    # Reset the character size.
symbol $1                     # Chose this symbol.
move 0 0                      # Move to the center of the panel.
dot                           # Draw the symbol.
end

limits -1 1 -1 1              # Set the limits.
set nsig 0                    # Use integer format.
set \10 -1                    # Initialize the loop counter.
loop 32 dosymbol \10 8 4      # Draw each symbol.
```

Figure 2 illustrates WIP's ability to handle images. One of two data sets used to generate this figure is an image from the VLA (FITS format) and the other from the BIMA array (Miriad format). The point of this figure (de Geus 1994) is to illustrate the ease at which two images with different resolutions and spatial extents can be overlaid.

Commands needed to generate Figure 2

```
levels -1 1 2 3 4 5 6 7 8 9 10 11 12 13 # Set the contour array.
slev A 25.0                  # Contours now at -25, 25, 50, etc.
ticksize 10 10 60 6          # Change the default tick intervals.
image s127.6cm               # Read in the VLA FITS image.
winadj 0 nx 0 ny             # Set the window's aspect ratio.
header rd rd                 # Read the image header and limits.
halftone 0.003 0.1           # Draw the halftone figure.
image s127.co 1 0.0          # Read the BIMA Miriad image.
header rd rd                 # Read the image header and limits.
contour                      # Draw the contour plot.
```

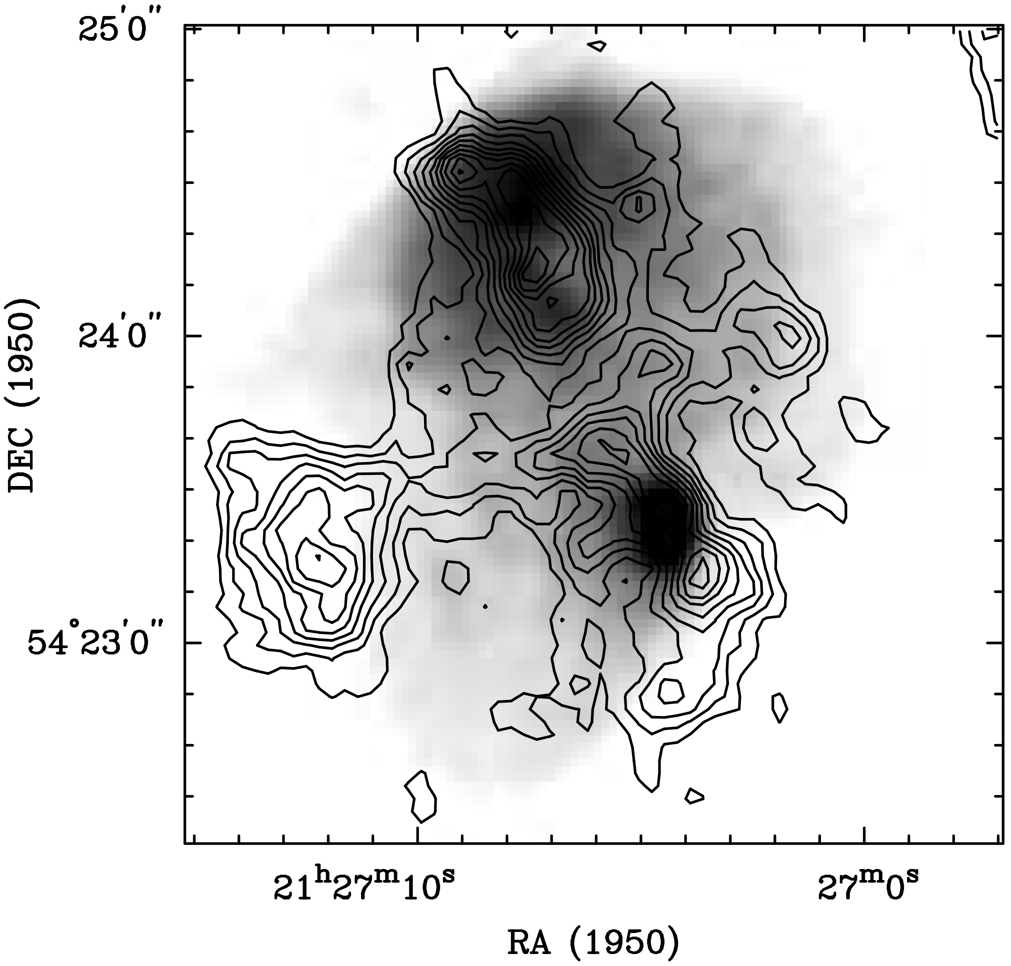

Figure 2. A 6 cm VLA image of the H II region S 127 overlaid with contours of the integrated CO emission observed with the BIMA array.

```
box bcnsthz bcnstvdzy       # Draw a box with RA/Dec labels.
xlabel RA (1950)
ylabel DEC (1950)
```

References

de Geus, E. J. 1994, in 3^{rd} Annual October Astrophysics Conference in Maryland, Back to the Galaxy, AIP Conference Proceedings, Vol. 278, eds. S. Holt and F. Verter (New York, American Institute of Physics), p. 481

Sault, R. J., Teuben, P. J., & Wright, M. C. H. 1995, this volume, p. **??**