

## What is DV?

The DV software is a tool used to view Fits data files. Brief summaries of its capabilities are:

- Manipulate fits images of up to 2048 by 2048 pixels using double float numeric types.
- Provides up to 8 buffers and 5 (or 7, or 9) canvases for holding and viewing data.
- Provide 8 graphical display formats: image, FITS header, histogram, line cut, X line cut, Pointer, Noise, and Stats.
- Performs various arithmetic operations on the fits data.
- Provides tools to calculate and view statistics on pixels.
- Produces postscript files of graphics for hardcopy output.
- Accepts data and command from other application using sockets.

DV is an X Windows application written using the GTK+ toolkit. It was developed by the NASA Infrared Telescope Facility, and is used as the facilities instrumentation FITS data viewer.

For more information, please visit the DV Home Page at <http://irtf.ifa.hawaii.edu/Facility/dv>. If you have questions or comments concerning DV, you may contact the author of the software via email at [denault@irtf.ifa.hawaii.edu](mailto:denault@irtf.ifa.hawaii.edu).

## Setting up your user account

Since DV is an IRTF in-house software application, it is already installed on all IRTF workstations. If you wish to have this application installed on your own workstation, please see the IRTF computer staff for more information. This section will cover the setup information needed on your user account to run this application. To setup your account, you will need to know where the application is stored on the hard disk. Normally the IRTF will install it in `/usr/local/dv`, however you should check with the computing staff on this. The directory `/usr/local/dv` will be used as an example in this document.

### 1) Setting the environment variable `$DV_HOME`.

Create a environment variable `$DV_HOME` to equal the name of the subdirectory where the application is stored on your hard disk. For example, a csh user can place the following line in the `.login` file to set `DV_HOME` to `'/usr/local/dv'`:

```
setenv DV_HOME /usr/local/dv
```

### 2) Startup configuration file `.dv-init`.

This step is optional, but highly recommended. When DV starts up, it searches for the file `.dv-init` in your home directory. If this file exists, DV expects it to contain DV commands. See DV Command Dictionary for the syntax. Using a text editor you can place instructions in this file to configure when it starts up DV. For example, the file may contain the following instruction to setup the directory paths and printer name.

```
Path $HOME/data
m.path $HOME/macro/dv
m.filemask *
printer irsubmitpr
```

## Starting DV

The DV program is a Xwindows application. You should be running an X server in order to used DV. Two methods are described below:

1. On the IRTF workstation, your user account should have a menu selection for DV on the workspace manage. Click on this to start DV.

2. To start DV from a xterm or shelltool window, type 'dv' in the window.

```
unix% dv &
```

### Command Line Arguments

DV can run on both 8-bit Pseudo Color or True Color (24 or 32 bits) displays. DV will try to execute using 8-bit pseudo color, then try true color. Sometime, DV is unable distinguish which modes are supported by the X Server and won't start. The following command line options can force DV to run in a particulate mode:

- P Pseudo Color. Tells DV to fun using the Pseudo Color Display mode. (This is the default)
- T True Color. Tells DV to run using the True Color Display mode. For example, this command forces DV to run in true color mode: `unix% dv -T`
- f fontName Specifies the default font. DV will attempt to use the named font as its default font. In this example, DV will use the 7x14 font: `unix% dv -f '7x14'`
- m Medium DV option – This option will cause DV to start with 7 canvases: 6 on the main window, 1-floating display. The default is 5 canvases: 4 on the main window, 1-floating display.
- l Large DV option – This option will cause DV to start with 9 canvases: 8 on the main window, 1-floating display. The default is 5 canvases: 4 on the main window, 1-floating display.
- p number Specifies a port number for DV's socket interface. A user may wish to use another port number besides the default 30123. For example, a second instance on DV would require a unique port number to support its socket interface. Legal value are from 30000 to 32000.
- R num
- C num
- S num – These options allow you to reconfigure DV's main window's display window. Normally you DV has 4 display window with each window having 256x256 pixel display. The range for R, C is 1 to 10, but the maximum number of display windows is limited to 15. The S range is 100 to 1024. For example,
  - 'dv -R 1 -C 2 -S 512 ' will provide a DV with 2 display windows (1x2), each 512x512 in size.
  - 'dv -R 3 -C 4 -S 150 ' will proved a DV with 12 display window (3x4), each 150x150 in size.

Sometimes the following message may appear in your console window.

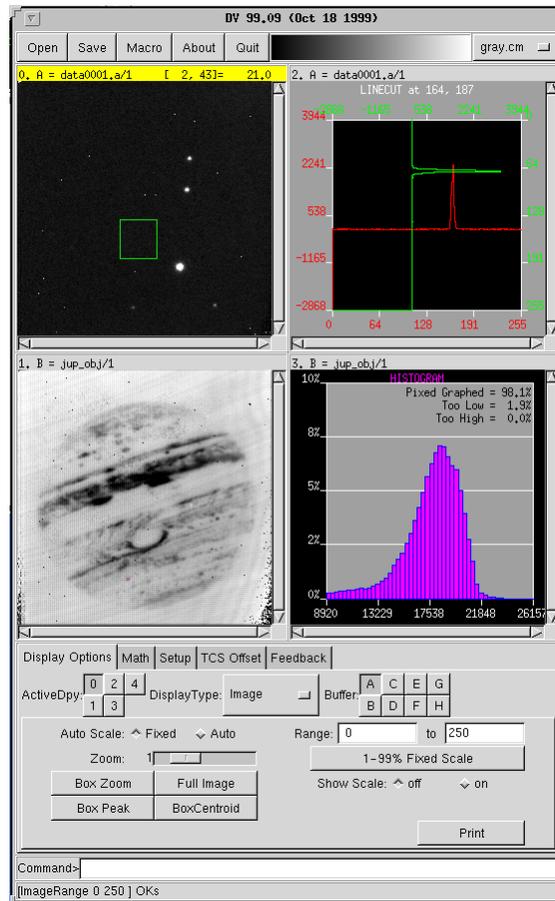
```
Run DV without Sockets? (y or n) <CR>:
```

If you are running DV with the instrumentation software, you must run DV with sockets. Otherwise you may run DV without sockets. When DV starts it tries to establish a socket to allow data and commands to be sent to it from the instrumentation program. This message can appear because:

1) DV is already running on the machine.

2) The operating system has not cleared the socket a previous version was using. If you just exited the program, it will take the operating system a few minutes to destroy the socket. During this period, you cannot establish another DV socket. Just wait 30 seconds and try again. (And maybe again).

## The Base Window



When DV starts, a window called the base window appears on your screen. Figure 1 is an illustration of the base window. The base window consists of the Menu Bar, 4 Viewing Canvases, a set of tabs to access DV's variety of parameters, a command line, and the Feedback Line.

Figure 1 – The Base Window

The **Menu Bar** - The menu bar and its button selections are shown below with a summary their purposes.



Figure 2 – Menu Bar

Description of Buttons - Here is a description of the buttons.

- Open – Brings up the Open File dialog box.
- Save – Brings up the Save File As dialog box.
- Macro – Brings up the Macro dialog box
- Quit – Quits the application.

The **Color Map Canvas** shows the current color map in use.

Four **Canvases** are located on the base window are 4 Canvases with horizontal and vertical scrollbars. They are used to display the data stored in DV's data buffers.

The **Main Panel** is used to set most options dealing with the image display and has several different view, which are detailed below.

The **Command Line** is used to enter commands manually.

The **Feedback Line**: The most recent command or error generated by DV is displayed here.

The Command Line and Feedback Line are discussed in the Main Panel section.

## The Canvas

The canvases are used to display the data. There are 5 canvases identified by the letters A, B, C, D and E. Canvases A to D are 256 by 256 pixels in size and are located on the base window. Canvas E is located in a floating window which can be re-sized. To bring up Canvas E, select its button on the 'Display Options' page. An example of a canvas is shown in figure 3.

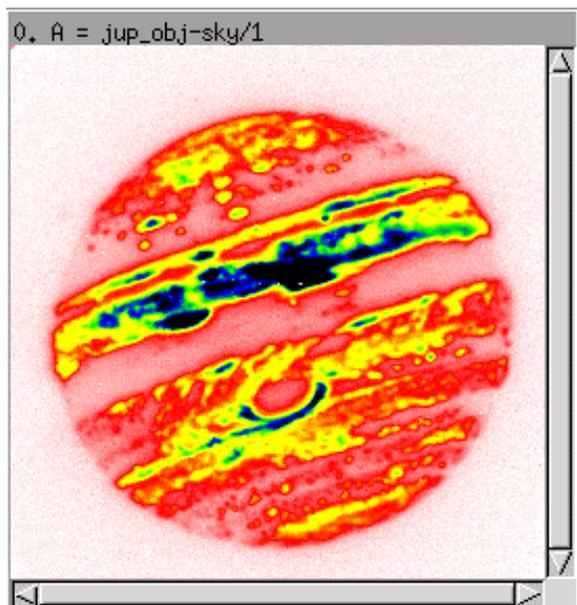


Figure 3

The **information area** shows some textual information about the data. The left hand side shows the buffer, the filename, and the divisor variable. For example, "B = jup\_obj-sky / 6" means this canvas is displaying the data in Buffer B, which contains the file "jup\_obj-sky", and that the data is represented at 1/6 of its true value. On the right side, additional information about the data or display is shown. This information depends on the display type and is described in Table A.

The **vertical and horizontal scrollbars** allow you to scroll the display if the image is larger than the canvas. Only the image and header display types can be scrolled. All other display types are scaled to the size of the canvas.

The **Canvas** is used to show a representation of the data according to the display parameters. In addition the canvas can capture mouse and keyboard events which can affect the display. See Table A – Description of Canvas Events for a description of the types of events for each display type.

Image Canvas Event	Action
Cursor Movements	Display the x, y location and value of the pixel in the information area. The text is usually in black. However, if the pixel is bad and the bad pixel mask is on, the text is colored red.
Dragging the cursor and pressing the menu (right) mouse button	This causes the colormap to be stretched and shifted. Vertical movements affect the stretch. Horizontal movements shift the color map.
Dragging the cursor and pressing the adjust (middle) mouse button	The dragging motion rubberbands a group of pixels. These pixels are known as the Object Box.
Dragging the cursor, pressing the adjust (middle) mouse button & holding down the shift key	Draws a line from the starting pixel until the dragging motion is completed. The endpoints of this line are copied to the 'From' and 'To' prompts in the TCS coordinates page.
Pressing the select (left) mouse button	Center the Object Box at the cursor's x, y location.
Press "L" in the image display	If the current active canvas is displaying a linecut from the same buffer as the image, pressing 'L' will set a new X, Y axis for the line cut.

Table A – Description of Canvas Events

## The Main Panel

The main panel contains several pages that allow the user to change display options, carry out math operations of the buffer data as well as computing statistics on the data. There are six main panel pages, Display Options, Stats, Math, Setup, TCS Offset and the Feedback page.

Selecting buttons or changing values on the various menu or panel items usually generates a command to DV to manipulate the data or change the display. These commands and the error messages from dv are displayed in the Feedback Page. In this way you can see the commands and responses generated by your actions.

The message output on the Feedback Page are color-coded as follows:

- Black – commands.
- Red – application warning and error messages.
- Blue – application messages.
- Gray – minor messages printed during execution.

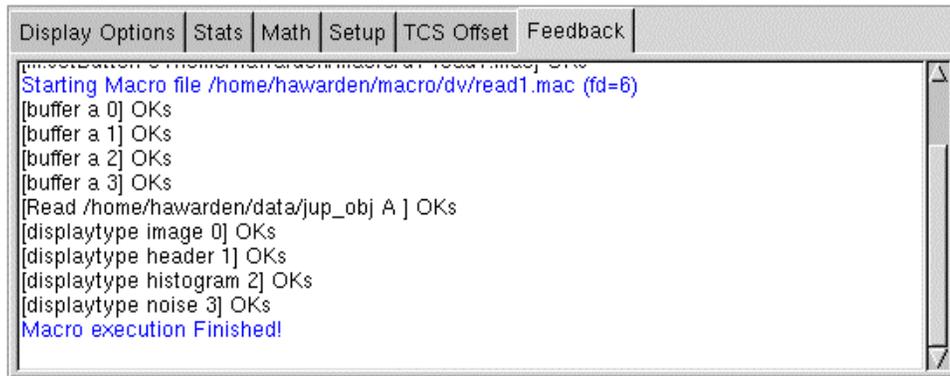


Figure 4 – The Main Panel

The panel items display in figure 5 will never change. These are described below.



Figure 5

**Command Line** - This text panel item allows you to type commands to dv. The dv commands are listed in the DV Command Dictionary.

**Feedback Line** - This text display item allows you to see the most recent command or error generated by DV. The messages are color-coded in the same fashion as the Feedback Page above.

## The Display Options Page

The display options page contains several panels that in turn contain buttons, prompts and menu items to control the display type and their options. For each canvas there is a display options page as shown below: Selecting the Display Type menu gives the following options: Image, Header, Histogram, LineCut, XLineCut, Noise, and Pointer.

The Display options page is divided into two parts. The upper part contains some general options: buttons to select the active display, an option menu to choose the display type, and another set of buttons which allow the user to associate different buffers with each of the display canvases. This part does not change.

However, the rest of the panel may change depending on the Display Type parameter. For each display type a different set of parameters is displayed. These panel items reflect the parameters of the current active canvas and display type. A description of the panel items for each display type is given below with an illustration of its panel.

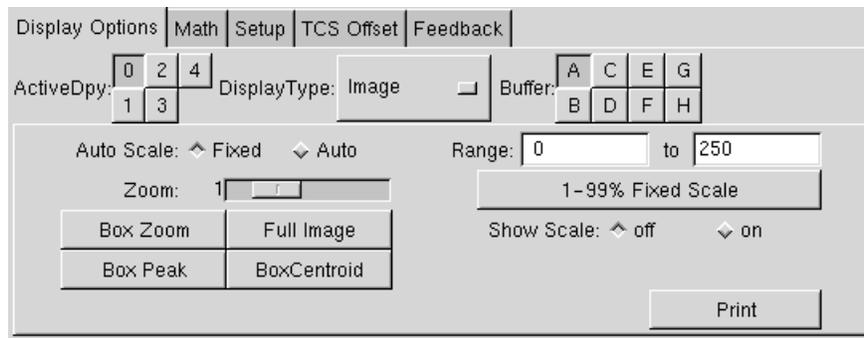


Figure 6 – The Display Options Page

**Active Display Buttons** – The selected button identifies the active canvas. The remaining panel items reflect the set up of the active canvas. Changing the values of any of the panel items only affects the active panel items.

**Display** – This option menu item selects the display type for the active canvas. The display types are Image, Header, Histogram, Linecut, XLinecut, Noise, Pointer, and Stats

**Buffer Buttons** – Each canvas will display the data from this buffer. You can view the data in a different buffer by changing this panel item.

The following describes each of the eighth display types available to the user.

1. The **Image** display has of the following panel items:

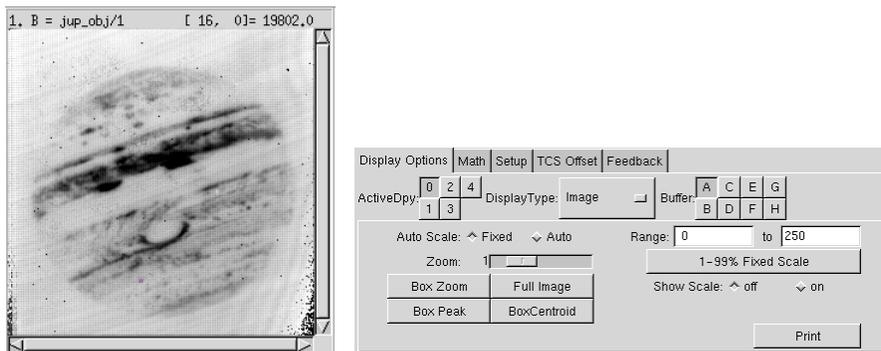


Figure 7 – The Image display & panel items.

**Auto Scale** – The scale options are fixed or auto. When auto is selected the Image Range is automatically adjusted to the minimum and maximum values of the data displayed in the canvas. This adjustment occurs whenever the data in the buffer changes. For fixed scale, the program will never change the Image Range.

**Zoom** – The zoom sets the zoom factor for the display. Adjust the scrollbar to the left (zoom out) or right (zoom in) to set the zoom to 1:1, 1:2, etc., data pixels per screen pixels.

**Box Zoom** – This button sets the zoom and adjusts the scrollbars of the canvas so that the Object Box is centered in the canvas and all the pixels within the Object Box are displayed at the highest zoom level.

**Full Image** – This button sets the zoom and adjusts the scrollbars of the canvas so that the entire image is centered in the canvas and all the pixels are displayed.

**BoxPeak** – This issues the BoxPeak command, which shows the peak pixel inside the object box.

**BoxCentroid** – This button issues the BoxCentroid command, which shows the Centroid position inside the object box.

**Range** – The color to the pixels are found by mapping the color map to the Header minimum and maximum values of the Image Range.

**1-99% Fixed Scale** - This button will execute an algorithm that determines the minimum and maximum range values such that 98% of the pixels is included within the range. 1% percent of the pixels will be above the maximum, and 1% of the pixel will be below the minimum.

**Show Scale** – Determines if an arcsecond scale is displayed along with the image. Note that the necessary information is also needed in the fits header or else the scale cannot be displayed.

**Print Image** – Select this button to get a hardcopy from the laser printer.

2. The **Header** display shows the text of the FITS header in the canvas.

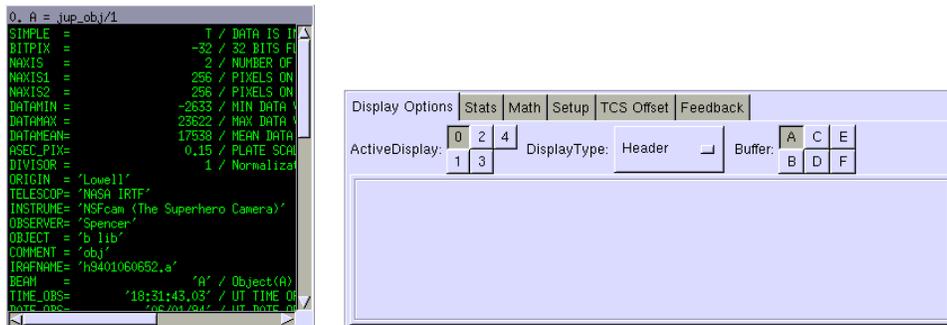


Figure 8 – The Header display & panel items

There are no options available for this display type.

3. The **Histogram** display draws a histogram on of the pixel's data distribution.

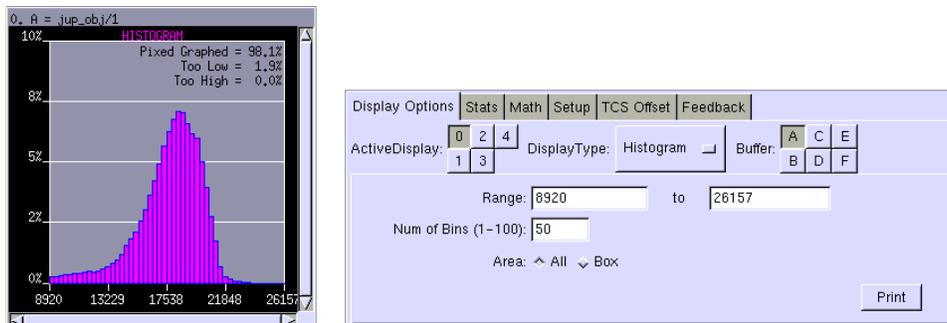


Figure 9 – The Histogram display & panel items

The Histogram panel consists of the following panel items.

**Range** – These values determine the minimum and maximum values of the X axis of the histogram, and will be the same as the range on the image display panel. Changing the histogram range will change the image range and vice versa.

**Num. of Bins** – This is the number of bars on the histogram. The size of the bin is determined by dividing the range by the number of bins.

**Area** – These controls allow you to indicate the pixels used in creating the histogram. You may use the entire image (All) or just the pixels specified by the ObjBox (Box).

**Print** – Select this button to produce a hardcopy of the histogram.

4. The **Linecut** display will show a graph of the pixel values along the x and y axes.

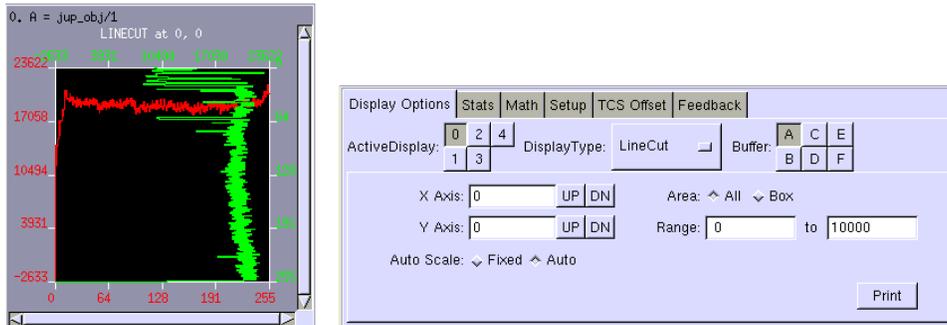


Figure 10 – The LineCut display & panel items

The Linecut panel consists of the following panel items.

**X Axis** – Identifies the X axis used to draw the line cut.

**Y Axis** – Identifies the Y axis used to draw the line cut.

**Auto Scale** – This switch determines whether to autoscale the graph or use Range to determine the scale.

**Range** – These values determine the minimum and maximum values of the scale when autoscale if off.

**Area** – The area along with the X & Y Axis variables defines the pixel domain for the graph. The options are All and Box. When All is selected, the entire canvas is plotted. When Box is selected, the Object box is used to define the region to be plotted.

**Print** – Select this button to produce a hardcopy of the linecut graph displayed in the canvas.

5. The **XLinecut** displays a single linecut graph between 2 points or pixels.

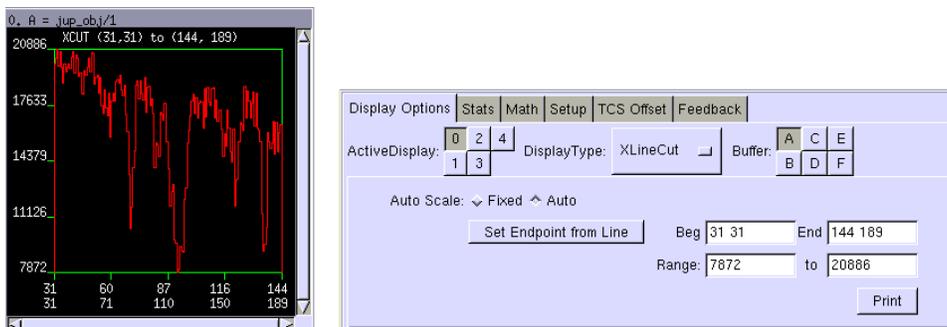


Figure 11 – The XLineCut display & panel items

The XLinecut panel consists of the following panel items.

**Set Endpoints from line** – Click this button to use the coordinates of a line drawn on the image as the endpoints of the graphs.

**Auto Scale** – This switch determines whether to autoscale the graph or use the range to determine the scale.

**Beg to End** – These coordinates identify the endpoints of the line of pixels used to construct the XLineCut graph. The Beg entry contains the x, y coordinates for the beginning of the line whereas the End entry contains the x, y coordinates for the end of the line. These can also be entered manually.

**Range** – When autoscale is off, these values define the scale on the data axis.

**Print** – Select this button to produce a hardcopy of the graph.

6. The **Noise** display is an engineering tool to display statistics on a group of pixels organized by columns.

The mod parameter indicates the number of output in the system. The min, max, mean, and standard deviation can then be displayed for each output. Graph 1 displays the Min, Max, and Mean of the pixels. The bar (I) shows the min and max while the point represents the mean pixel value. Graph 2 display the standard deviation for each mod.

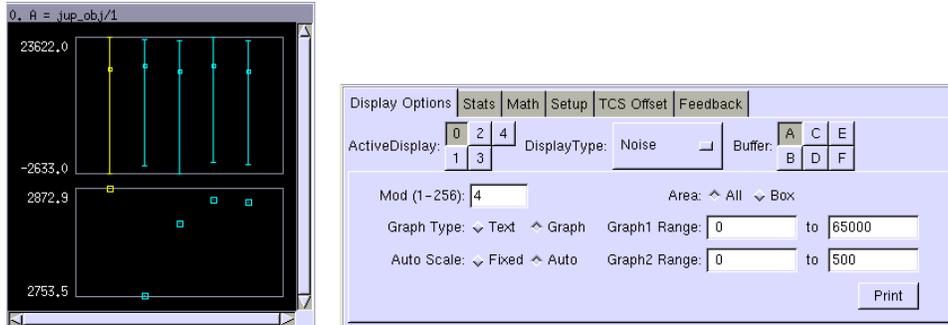


Figure 12 – Noise display & panel items

The Noise panel contains the following panel items:

**Mod** – control how columns are group together. For example, 4 would group every 4th column giving stats on 4 different groups of pixels.

**Graph Type** – This control allows you to indicate the display mode for the graph. ‘Graph’ is shown above. ‘Text’ gives a collimated printout of the min, max, mean, and standard deviation.

**Area** – This control allows you to indicate the pixels used in creating the graph. You may use the entire image (All) or just the pixels specified by the ObjBox (Box).

**Auto Scale** – The Y axis is automatically scaled when on, or scaled using the range values when off.

**Graph 1 Range** – When autoscale is off, these values define the Y scale for the Min, Max, Mean graph.

**Graph 2 Range** – When autoscale is off, these values define the Y scale for the standard deviation graph.

7. The **Pointer** display is a combination of several other displays.

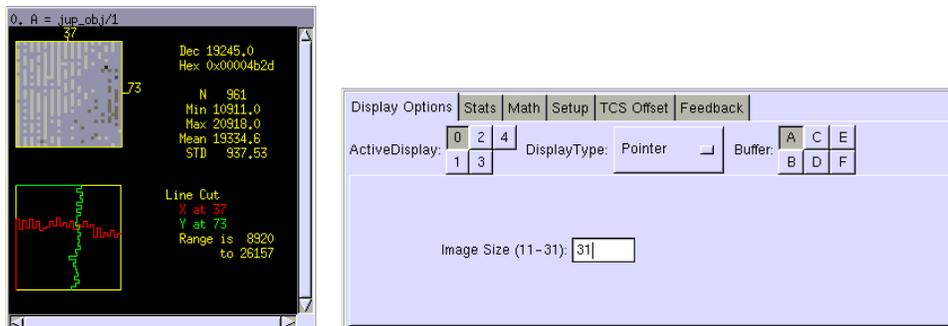


Figure 13 – Pointer display & panel items

The Pointer display contains three areas: a zoomed image display, an XLinecut display, and some textual information about the data. The image display box is activated when the mouse is moved onto an image displayed in one of the other canvases. If no image is displayed, nothing will be shown here. The XLinecut box displays a normal XLinecut for the pixel over which the mouse is hovering. The text area displays statistics about the image box.

The Point Panel contains the following panel items:

**Image Size** – controls the height and width (in pixels) of the image box.

8. The **Stats** display provides elementary statistical information on groups of pixels.

The information area displays the sum, mean, and standard deviation of the data display in a canvas. The mean, variance, and standard deviation for the entire image are displayed on the top row. Below this are 3 columns labeled 'Object', 'Sky', and 'Object-Sky', with statistical values for each column. The object column displays the value on the object box. The sky column displays statistics on the sky box. If the object and sky box are of equal dimension, the object-sky column will display statistics on the pixel values obtained by subtracting the object box from the sky box.

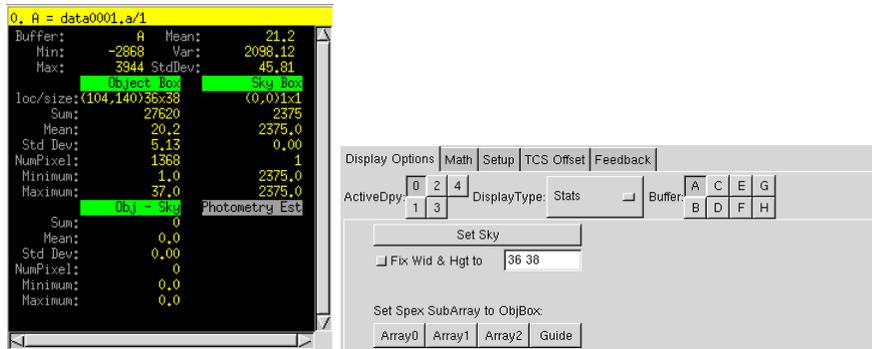


Figure 14 – Stats display & panel items

The Stats Panel contains the following panel items:

Selecting the **Set Sky** button sets the location and size of the sky box to be equal to the object box.

To set the width and height of the subarray type the width and height in the **Fix Wid & Hgt** to text entry field. The checkbox, when select, caused to the wid&hgt to be fixed.

The **Spex Subarray** buttons allow a user to set the size and position of the subarrays used in the various modes of SPEX. The position and size of the ObjectBox is sent to SPEX's SubArray 0, 1, 2, or Guide Array, when you select the **Array0**, **Aarray1** or **Array2**, and **Guide** buttons.

## The Math Page

The Math Page allows 4 types of operations: 1) Arithmetic on the pixel values, 2) Transferring data between buffers, 3) Deleting data from a buffer, and 4) Rotating an image. Note that there are buttons labeled 'Execute', 'Copy', 'Clear' and 'Rotate'. Near these buttons are various panel items used to identify the buffer or other parameters for each operation.

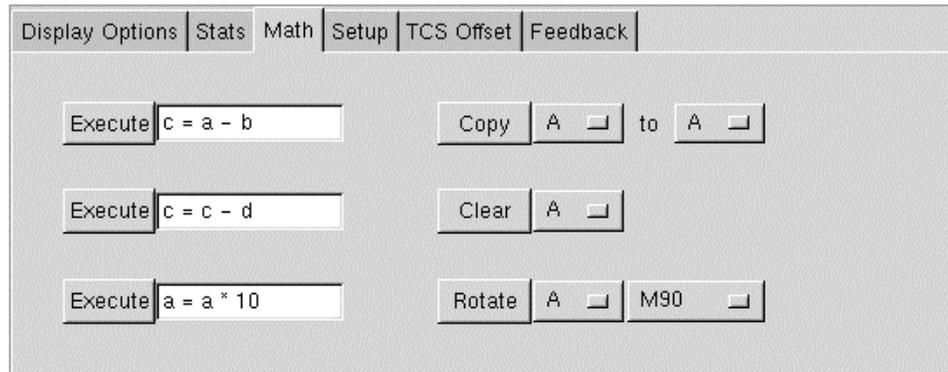


Figure 14 – The Math Page

For buffer Arithmetic, form an expression by selecting a destination buffer, operation (+ - \* /), and the operands. The first operand is always a buffer index, the second can be a buffer index or a constant. Two examples of math expressions are shown below:

'c = a - b' will subtract buffer b from buffer a and store the results in buffer c.

'c = c \* 100' will multiply every pixel in buffer c by 100 and place the results in buffer c.

When you are done forming your expression, selecting the **Execute** button beside the entry area to perform the operation.

To **copy** data from one buffer to another use the top row of items on the right side of the panel. First select the destination and source buffers. Select the **Copy** button to perform the transfer.

To **clear** data from the buffer, identify the buffer by selecting it to the right of the clear button, then click the **Clear** button to delete the data.

The **rotate** operation allows you to rotate the data to the left or right. Select the desired buffer, then click the **Rotate** button to rotate in the selected direction. 'M90' rotates the image -90° (counterclockwise), 'P90' rotates the image +90° (clockwise), and '180' rotates the image 180°.

## The Set Up Page

The Set Up Page contains panel items which are not likely to be changed very often. The items are usually self-explanatory. For a detailed explanation of each option, refer to the DV Command Dictionary.

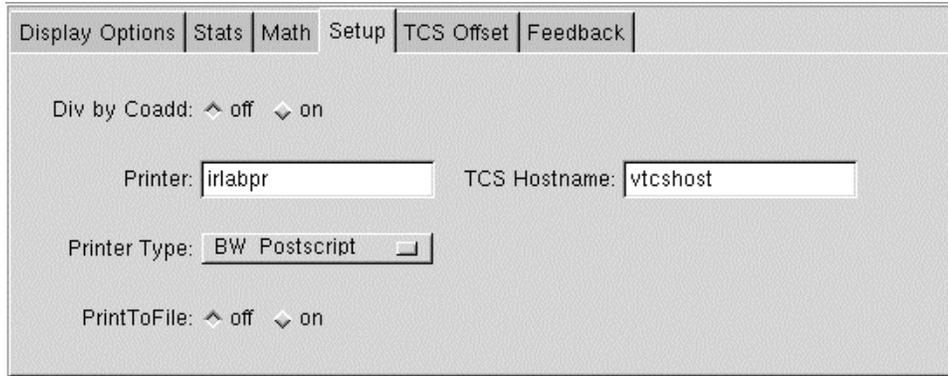


Figure 15 – The Set Up page

## The TCS Coordinates Page

The TCS Coordinates page allows you to calculate the relative offsets in RA and DEC between 2 pixels in an image. This offset can be transmitted to the Telescope Control System, thus providing a convenient way to position objects on the array.

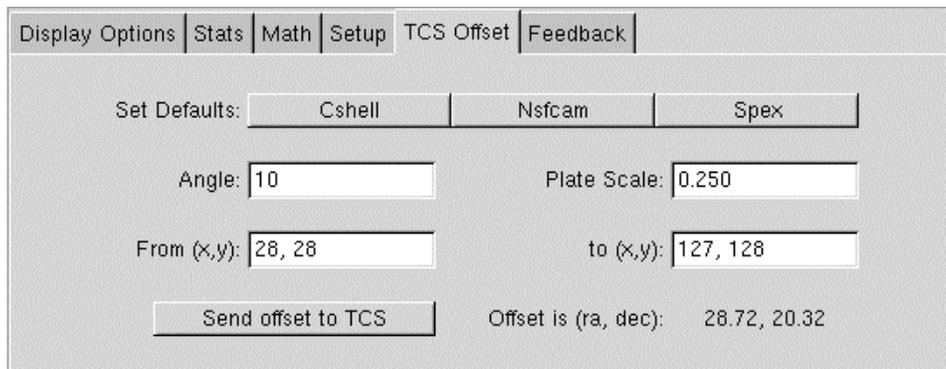


Figure 16 – The TCS Coordinates page

The following steps outline the procedures for offsetting the TCS.

1. Enter the **Angle** and **PlateScale** information. This information varies with the array and instrument used. The following table contains the value for the IRTF Facility Instruments. Please confirm these values with the Telescope operator during your run.

Instrument	Angle	PlateScale
CSHELL	270	0.2
NSFCAM	0	0.3
SPEX	10	0.25

The **Set Defaults** buttons set the default values for CSHELL, NSFCAM, and SPEX.

2. Identify the **From** and **To** points in terms of their x, y pixel location on the image. This can be accomplished by the following method:

- a) Manual type in the coordinated values.
- b) Drawing the XOR line on the image. First, place the cursor in the image at the **From** pixel location. While pressing the SHIFT key, press the middle mouse button and drag the cursor to the **To** location. While you are dragging the mouse a line should be drawn between the two points. The act of drawing this line updates the **From** and **To** coordinates values in the TCS Coordinates page.

3. The **Offset is (ra, dec)** are the offset coordinates. These are calculated automatically as you move the mouse.

4. If you agree with the offset coordinates in the **Offset is (ra, dec)** display, press the **Send offset to TCS** button. Pressing this button will sent a command to the TCS to offset its position by the indicated coordinates.

Note: The TCS command issued by DV to perform the offset is "0 ra dec 0 C.SCN".

## The Open File Dialog Box

Selecting the File button on the menu bar causes the **Open File** dialog box to appear on the screen. This dialog box enables you to read data files into the data buffers.

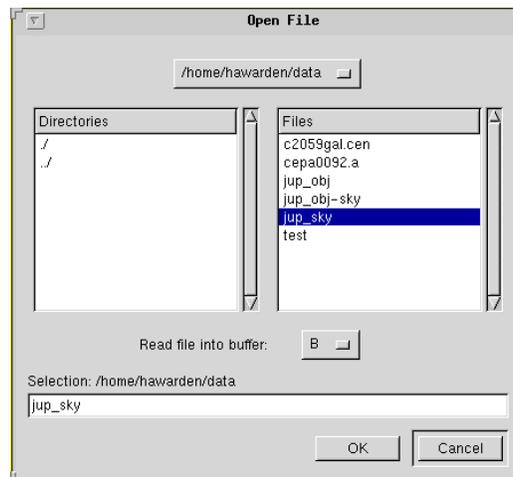


Figure 17 – The Open File Dialog Box

There are two lists on this window. On the right is the file list, which lists all the files in the Path matching the Filemask string pattern. You may select filename with the mouse to identify the file you wish to read. On the left is the directory list. The directory list display the name of the directory in the current path. Using the mouse you can double click on these name to change the Path.

The other panel items inside this dialog box are:

**Directory Option Menu** – This option menu displays the directory history from the root directory down to the one you are currently looking at.

**Buffer** – The data will be read into the buffer identified by this panel item. Select the desired buffer from the menu.

**Selection Entry Line** – The entry line allows you to type in the path and/or the filename of the file you want to open.

**OK** – Selecting this button will cause the selected file to be opened into the chosen buffer.

**Cancel** – Selecting this button hides the dialog box.

## The Save File As Dialog Box

Selecting the Save button on the menu bar will cause the Save Data dialog box to appear on the screen. This dialog box enables you save or write the data from a buffer to a file.

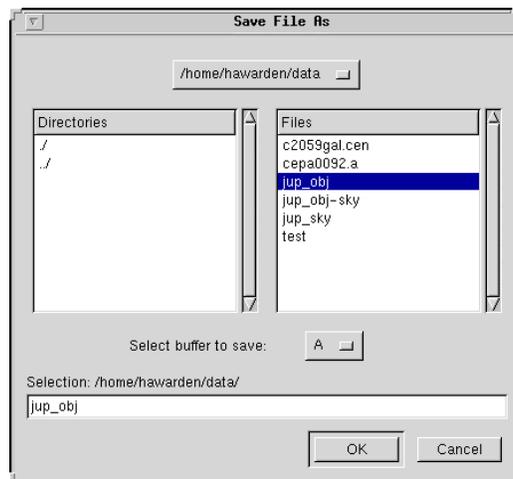


Figure 18 – The Save Data Dialog Box

The other panel items inside this dialog box are:

**Directory Option Menu** – This option menu displays the directory history from the root directory down to the one you are currently looking at.

**Buffer** – The data from the buffer identified by this panel item will be saved to the file.

**Selection Entry Line** – The entry line allows you to type in the path and/or the filename of the file you want to save to.

**OK** – Selecting this button will cause the chosen buffer to be saved into the selected file. If the file already exists, it will be overwritten. If it does not, a new file will be created.

**Cancel** – Selecting this button hides the dialog box.

## The Macro Dialog Box

DV provides a way of creating and executing Macro files. A macro file is a text file containing commands. There should be one command per line and the syntax for commands is defined in the DV Command Dictionary. On the menu bar, the Macro button brings up a dialog box that allows you to create, edit, and execute macro files. This dialog box contains a text display window, a file list window, and a group of panel items.

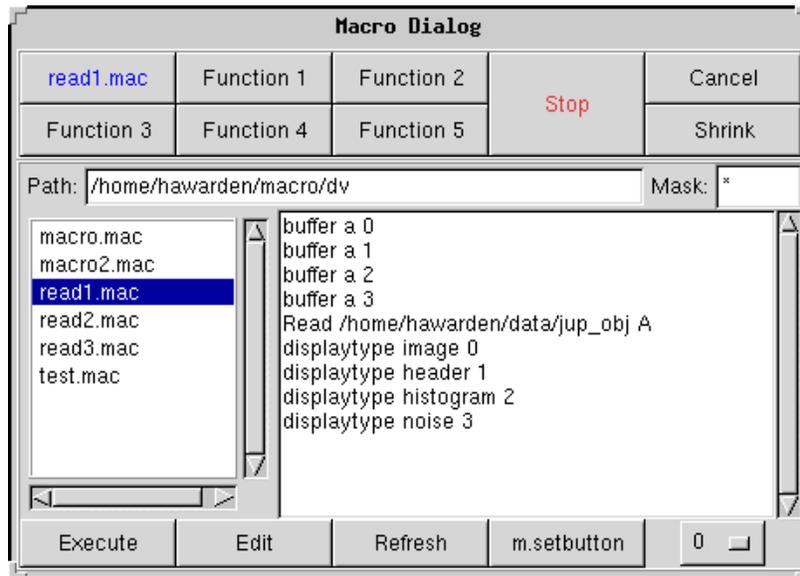


Figure 19 – The Macro Dialog Box

The following panel items will help you manipulate macro files:

**Six Function Buttons** – These six buttons can be associated with macro files. After the button has been linked to the file, clicking the button executes the macro. The name of the associated file is displayed on the button.

**Stop** – This button stops the currently executing macro.

**Cancel** – This button hides the dialog box.

**Shrink/Expand** – This button hides or shows the lower part of the dialog box.

**Path** – This text panel identifies the directory where your macro files are stored.

**Mask** – The mask is a regular expression used with the path to determine the names of the files displayed in the file list. Selecting **Refresh** will update the list.

**Execute** – This button executes the chosen macro.

**Edit** – This button opens the selected file using the word processor 'NEdit'. The file can then be edited, saved, etc.

**Refresh** – This button updates the file list.

**m.setbutton** – Use the option menu next to this button to select the button you want to associate with the selected file. Clicking this button associates the file with the function button. When a file has been associated with a button, the name of the file is displayed on the button.

To execute a macro file, select the file in the list then click on the **Execute** button. DV will read the file and execute each line. The commands and their error messages ('Macro execution Finished' when successfully executed) will be written to the Feedback Page. If you wish to abort a macro file in progress, click on the **Stop** button.

## How to display a Gaussian fit to dv XLineCut display.

It is possible to display a Gaussian fit to the line cut displayed by XLineCut. This is done by exporting the Y data values used to produce the cut. Then using the **xfit** program, display the data, and fit the graph.

### Account Setup information:

The environment variable XGFIT\_HOME need to be set to /usr/local/bin in order to use xffit on the IRTF workstations. Adding this line to your .cshrc file should work:

```
setenv XGFIT_HOME /usr/local/bin
```

### Procedure to product a gaussian graph:

1. Use XLineCut to select the data values your are interested in using for the Gaussian fit.
2. Press the "Output Gaussian Data" button. This will created a text file name /tmp/xffit.dat on your computer. This data file contains the data points used to produce the XLineCut graph.
3. Start the xffit program (illustrated below) by clicking on the workspace menu xffit (under the NSFCAM menu) or type xffit from the command line:

```
% xffit
```

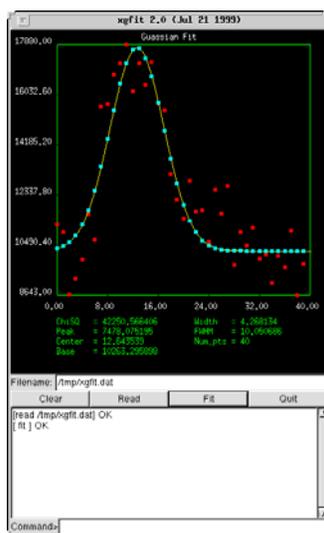


Figure 20 – Gauss

Click the **Clear** button to clear the data from memory.

Click the **Read** button to read the data file.

Click the **Fit** button to calculate the fit.

### What if it doesn't work?

Calculating a Gaussian fit is a very difficult procedure. The xffit program easily crashes if the data is not easy to work with. If xffit crashes when trying, there are some suggestions:

1. Redraw the XLineCut line and create a new xffit.dat files. Try not to select a line cut which contains bad pixels and an unusual data value which could throw off the fit.
2. Negative numbers are a known problem. Using data from subtracted image can cause xffit to crash. Adding an offset to the data to eliminate negatives may help.
3. You can also edit the file /tmp/xffit.dat to remove the abnormal or 'invalid' data point to help the xffit program work. Data sets with erratic data points can also cause xffit to crash. Smoothing the data set will allow xffit to work properly and calculate the correct FWHM value.