Part I: Archive Retrieval Tool: arr User's Guide

Carl Timmer (timmer@atdiv.lanl.gov)
EPICS R3.12
February 10, 1995

EPICS software was produced under U.S. Government contracts:
(W-7405-ENG-36) at the Los Alamos National Laboratory, and
(W-31-109-ENG-38) at Argonne National Laboratory.

Initially developed by:
 The Controls and Automation Group (AOT-8)
 Ground Test Accelerator
 Accelerator Technology Division
 Los Alamos National Laboratory

Co-developed with:
 The Controls and Computing Group
 Accelerator Systems Division
 Advanced Photon Source
 Argonne National Laboratory

CHAPTER 1 Getting Started

1.1 General Background

In the course of running an experiment with EPICS, data is often archived (stored in files) using either the **AR** or the **AR_cmd** program. The difference between these programs is simply that **AR** provides a graphical user interface while **AR_cmd** uses text commands. The resultant stored data can be viewed by either the **ARR** or the **arr** program. Both of these programs are graphical in nature. **ARR** is the original archive viewer but unfortunately uses SUN-specific libraries, and so it was thought prudent to rewrite this program in a platform-independent manner.

The new archive viewer, **arr**, is written in tcl/tk and runs on virtually all UNIX systems. The user is also sure to find **arr** much easier to use and many of the **ARR** bugs gone. In addition, tcl/tk is simple enough so that **arr** can be changed without too much pain by intermediate level programmers. For details on programming, see Part II of this manual which is the “arr Programmer’s Guide”.

1.2 Function

There are a number of **arr** capabilities provided:

1. selection of binary and ascii archive files with wildcard file listing
2. help window for each top level **arr** window
3. one-step selection of process variables with wildcard viewing
4. data retrieving options
5. one-button loading of data
6. textual viewing of timestamps, data, and status codes with font selection

7. plotting of numerical data with any number of graphs
8. fully configurable graph parameters
9. color postscript output to files and/or printers of graphs and text

1.3 Starting the archive viewer

The archive viewer is started by simply typing `arr` at the UNIX prompt. See the "arr Programmer's Guide" to find a list of system hardware and software requirements for running **arr**.

1.4 Manual Nomenclature

References to: program or file names are in **bold type**; window, button, menu, or other widget names are in *italic type*; keyboard input are in `Courier type`; and document names are in quotes.

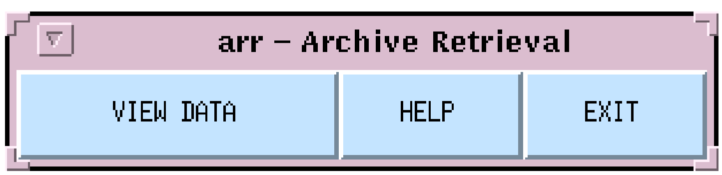
CHAPTER 2 Data Retrieval

This chapter leads the reader through all the steps of loading data before viewing.

2.1 Top Level Window

The window that appears upon typing `arr` can be seen in Figure 1 immediately below.

FIGURE 1 top level window



2.1.1 Exit

The *EXIT* button quits the application.

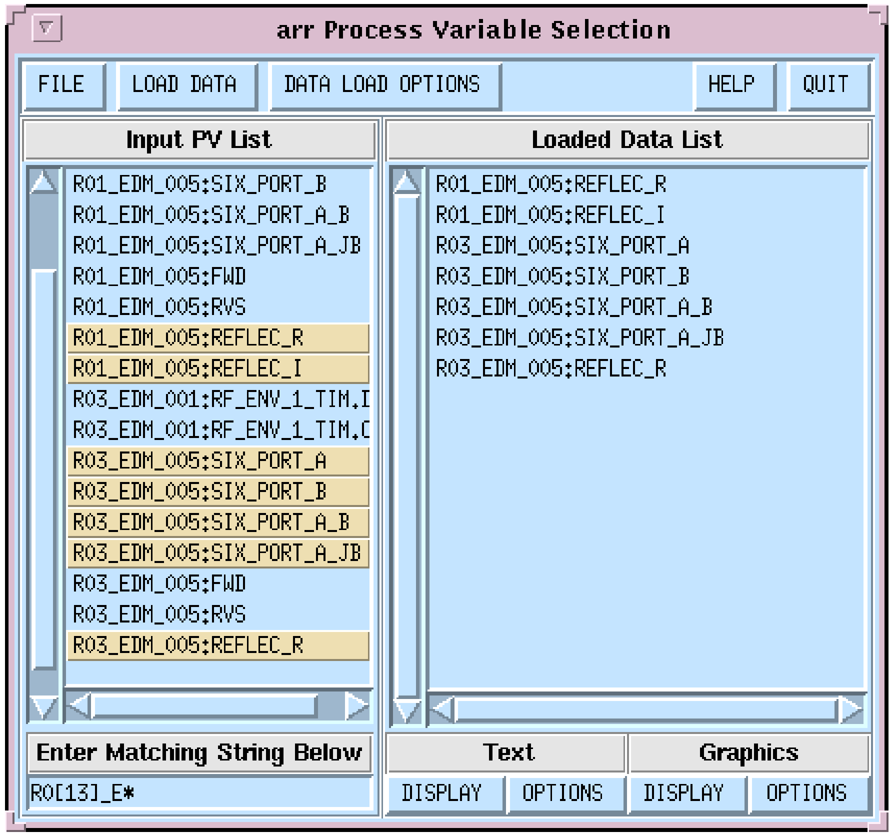
2.1.2 Help

The *HELP* button calls up a window describing the function of all the buttons. (Much as we're doing right now).

2.1.3 Viewing Data

The *VIEW DATA* button displays the window used to retrieve data into the archive viewer and brings one "into" the application (see Figure 2).

FIGURE 2 data retrieval window



2.2 Process Variable Selection Window

The *Process Variable Selection* window is called up by selecting the *VIEW DATA* button from the application's main window. It is used to: select a data file, select data loading options, and load data. It also allows for launching windows to display text and graphic data, or to display their respective options. (Incidentally, for those readers with color postscript printers, all the figures in this document are in color).

2.2.1 File Selection

The *FILE* button, in the upper lefthand corner, brings up a window that allows one to chose a file from which to read data. When a file has been chosen, **arr** reads it for a list of all available process variable names which it then places in the *Loaded Data List* and it also finds the beginning and ending timestamps. See Figure 3 for more details.

The directory from which one selects the file is carried through invocations of the *FILE* button but not through successive invocations of **arr** itself. The default directory can be set by simply modifying the **arr** script. See section 2.3 for more details.

2.2.2 Wildcard PV Listings

For a particularly long list of pv's (process variables) it may be easier to view a partial listing in the *Input PV List*. To do that, one needs only to type a string into the entry space below the label *Enter Matching String Below.* Only the pv's which match this string will be shown in the list. The "glob-style" rules are used for pattern matching. For those unfamiliar this system, the following list is all that is necessary to know:

1. * matches any sequence of zero or more characters
2. ? matches any single character
3. [*chars*] matches any single character in *chars*. If *chars* contains a sequence of the form a-z, any character between a and z, inclusive, will match.
4. \x matches the single character x. This allows matching of characters like * ? [ ] \ .

The example in Figure 2 shows "R0[13]_E*" which matches pv's starting with "R01_E" or with "R03_E". The default is to show all pv's.

2.2.3 Data Loading Options

The *DATA LOAD OPTIONS* button calls up a window allowing configuration of the data to be loaded. Starting and ending criteria, discarding of partial samples, rounding timestamps, and conditional acceptance of samples can be set here. These options must be set before data is loaded. See Figure 4 and section 2.4 for more details.

2.2.4 Selecting & Loading Data

After a file is chosen and the data load options are set, selecting which pv's one is interested in is the next step. The highlighted areas in Figure 2 give an example of pv's which were selected to have their data loaded. By default they also appear in the *Text Display List* so as to automatically be seen in the text data display.

The selection of pv's is done with the mouse in the following manner. To select a single pv, click on it with the left mouse button. To select a continuous range of pv's, hold the left button down and drag the mouse over the desired range of pv's. Another way to achieve a range of selected items is to click the left button on 1 pv, move the mouse to another pv and hit shift-left button. This selects the entire range between the 2 pv's. If one desires a disjointed selection, the simplest way is to use control-left button to select individual pv's without undoing previous selections. A second way of producing disjointed selections is to select a large block then use the right button to deselect unwanted pv's.

Actually loading the data into memory is accomplished by depressing the *LOAD DATA* button. Once data is loaded, it can be reloaded (if for example the loading options are changed) by simply hitting the *LOAD DATA* button without having to so any selecting.

2.2.5 Textual Display of Data

The process variables that have been selected can be viewed by pushing the *DISPLAY* button under the *Text* heading. This brings up a window which shows the pv names, timestamps, data, and data status codes. All data is displayed by default. The *OPTIONS* button under the *Text* heading is used to change fonts and also to make a hardcopy of or export the data. For more detail see Figure 5.

2.2.6 Graphical Display of Data

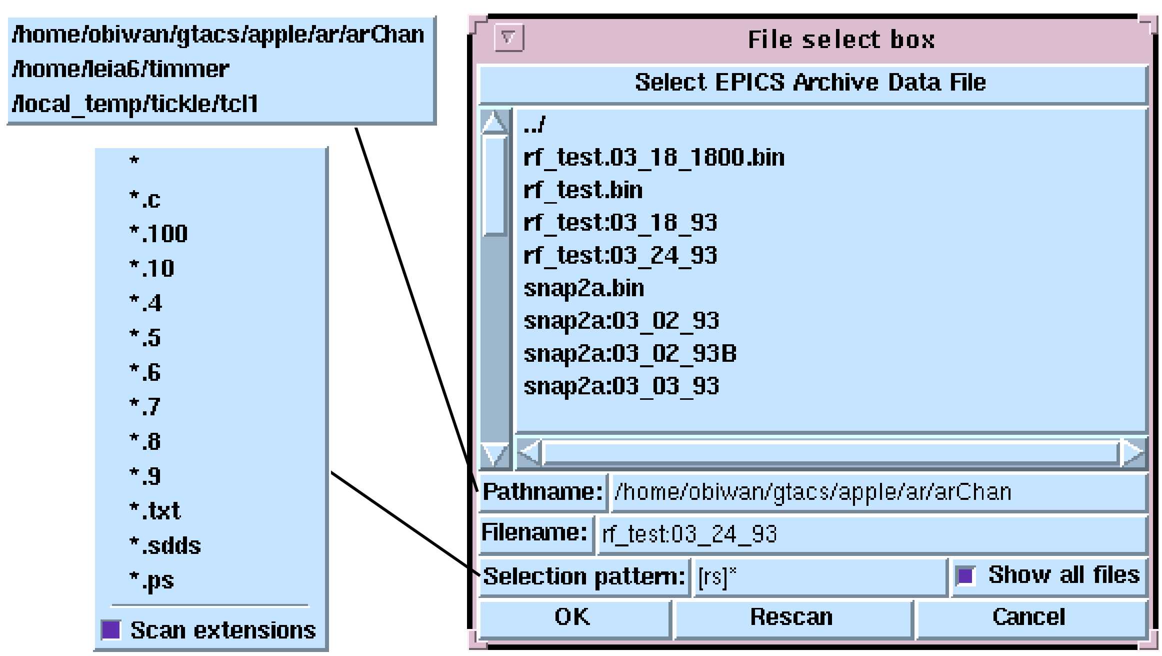
The process variables that have been selected and are numerical in value can be used in graphs specified by the user. By default, when starting up arr, the first 8 channels whose data are loaded are automatically plotted in a graph. Once graphs have been created, they can be viewed by pushing the *DISPLAY* button under the *Graphics* heading. Initially each graph is blank but can be configured by use of the *Graphics Display Options* window seen by pressing the *OPTIONS* button under the *Graphics* heading. The features of each graph under user control include things like: pv's (up to 8 per graph), colors, symbols, symbol sizes, legend, line dashing, line width, titles, tick marks, axis type, and scaling. For more detail see Figure 11.

2.3 File Selection Window

The selection of data files is accomplished by hitting the *FILE* button of the *Process Variable Selection* window (see Figure 2). This action calls up the *File select box* window as seen in Figure 3 following.

Capabilities of the file selection window include: keeping all visited directories in a list for fast access, wildcard file name viewing, showing/hiding files beginning with ".", rescanning the directory, and scanning file names for extensions which may be automatically used in the wildcarding.

FIGURE 3 file select window



2.3.1 Path Names

The file's path can be typed manually into the entry box next to the *Pathname:* menubutton (see Figure 3). Clicking the mouse on the menubutton shows a list of all the directories visited so far, allowing an easy return to previous items. Again, Figure 3 shows the resulting menu. Weaving up and down in the directory structure can also be done through mouse work in the file listing.

2.3.2 File Names

File names can be typed manually into the entry box next to the *Filename:* label. They may also be selected with the mouse in the file listing. Hitting the *OK* button or double-clicking with the mouse on the file name will make the selection final.

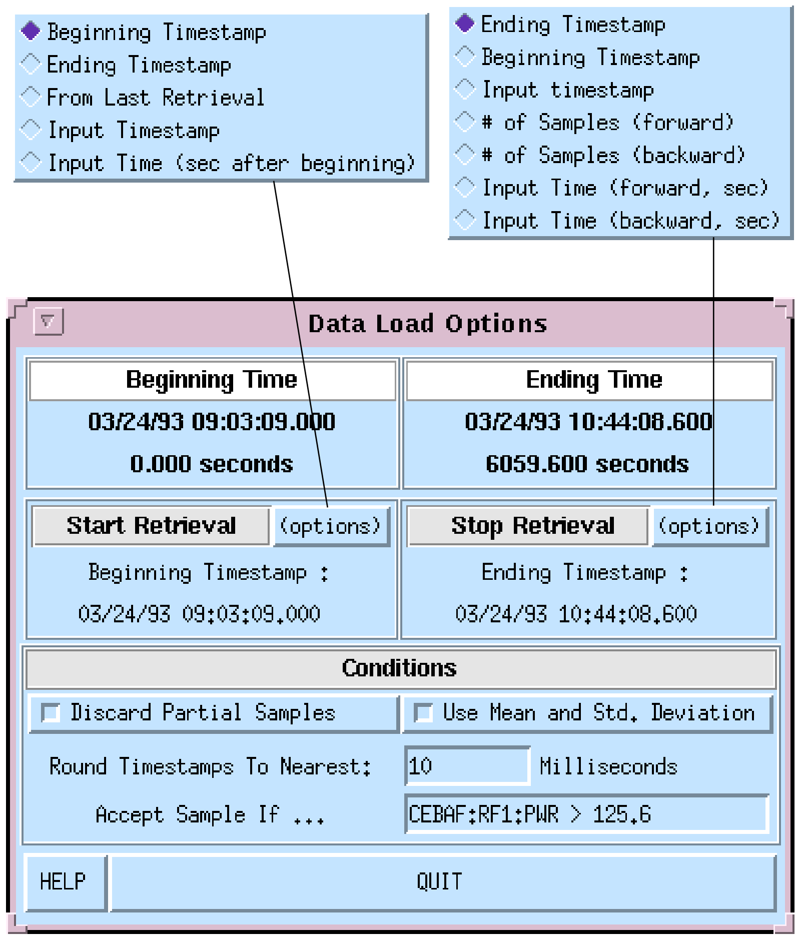
2.3.3 File Name Viewing

There are a number of options useful in determining which file names are viewed in the listing. One may simply type in a "glob-style" string (see section 2.2.2) in the entry box next to the *Selection pattern:* menubutton in order to do wildcard matching. Hitting the menubutton itself will allow one the option of having file extensions automatically scanned to produce a listing which then becomes part of the menu. Figure 3 shows what the resulting menu looks like. These extensions options can be used to wildcard the file name listing. Additionally, the *Show all files* checkbutton will show all files beginning with "." if selected. Finally, the *Rescan* button will reread directory contents ensuring that the file information is up-to-date.

2.4 Data Loading Options

There are a number of things to consider when retrieving data from a file. One may consider: starting and stopping conditions, partial samples, rounding timestamps, and logical conditions. All of these options can be set by pressing the *DATA LOAD OPTIONS* button in the *Process Variable Selection* window (see Figure 2). This action brings up the *Data Load Options* window as seen in Figure 4.

FIGURE 4 data loading options window



2.4.1 Retrieval Starting Point

There are 5 options for specifying the starting condition during a retrieve. In Figure 4 above, notice that a menu allows a choice between the beginning or earliest timestamp, the ending or latest timestamp, the latest timestamp of a previous retrieval from the same file, a user-given timestamp, or a time in seconds past the beginning timestamp.

2.4.2 Retrieval Stopping Point

There are 7 options for ending a retrieval: the ending or latest timestamp, the beginning or earliest timestamp, a user-given timestamp, the number of samples desired forward in time, the number of samples desired backward in time, a time in seconds past the starting point, or a time in seconds before the starting point. There is a maximum limit of 10,000 data points.

Of course there are some restrictions on which combination of options are available. One cannot specify an stopping point earlier than the earliest timestamp nor a starting point later than the latest timestamp. **Generally, the quickest way to retrieve data is to specify the number of samples forward in time as the stopping condition.** The default is to retrieve the latest 24 hours worth of data. In other words, the default is a starting condition of *Ending Timestamp* and a stopping condition of *Input Time (backward,sec)* set to a value of 86400 seconds.

If no file has been chosen at the time the starting or stopping menubuttons are invoked, a warning is sent to the user to "Choose a File First". Likewise, if no data has yet been loaded from a selected file, the third starting option - *From Last Retrieval* - will tell the user to "Load Data First".

2.4.3 Discard Partial Samples

Partial samples are data samples at one particular timestamp in which only some of the process variables have data. These data sets can be ignored by selecting the *Discard Partial Samples* option.

2.4.4 Use Mean and Std. Deviation

If a snapshot contains more than 1 sample (each of which can contain many process variables), then it's possible to retrieve the mean and standard deviation of each process variable. There will be one mean and standard deviation pair per process variable for each snapshot.

2.4.5 Round Timestamps To Nearest

By selecting the *Round Timestamps To Nearest* option and entering the desired time in milliseconds into the entry box to the right, one can round the timestamps found in the data file.

2.4.6 Accept Sample If ...

By selecting the *Accept Samples If* ... option, logical conditions can be placed on the retrieval of data. In other words, data is retrieved only if the condition is met. The condition is entered in the box to the right of the appropriate checkbutton. The form of the condition must follow the format shown here:

```
pvname operator value
```

where `pvname` is, of course, the name of a process variable in the file of interest; `operator` is a relational operator such as >, >=, <, <=, == (equal to), != (not equal to); and `value` is the value of the process variable.

CHAPTER 3 Textual Viewing Of Data

This chapter leads the reader through the means of viewing data in textual form and printing those data to both printers and files.

3.1 Text Display Window

3.1.1 Regular Data

Once the data is loaded from an archive file, one can view it textually by pressing the appropriate *DISPLAY* button in the *Process Variable Selection* window (see Figure 2). The resulting window will look something like that of Figure 5. A blue header contains the name of all the process variables and their units (if any). Time stamps appear on the left with the data in columns under the appropriate names. On the right side of each data point is a 2-character data status code. The first character of that code is always printed out while the second is optional. A table of the codes are given in Table 1.

FIGURE 5 text display window for regular data

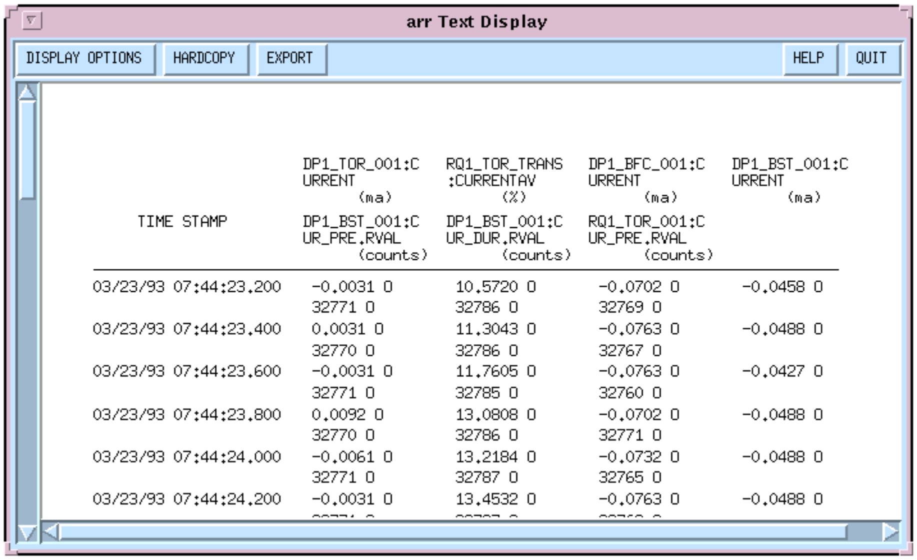


TABLE 1 data status codes.

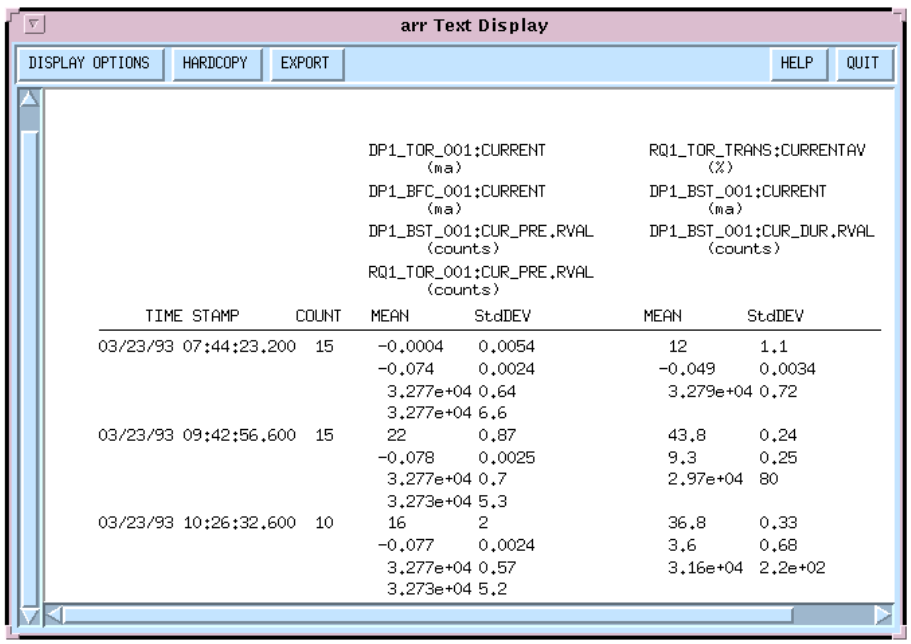
Type	Character	Description
1st character	O	okay, there still might be an alarm
“	R	okay, it restarts a data run
“	F	okay, it’s value is filled in from that of a prior proper timestamp
“	M	no value with a proper timestamp
“	S	okay, it starts a snapshot
“	I	value is invalid (alarm severity is INVALID)
2nd character	blank	ordinary value
“	R	hardware read alarm
“	H	highhigh alarm limit violated
“	h	high alarm limit violated
“	l	low alarm limit violated
“	L	lowlow alarm limit violated
“	D	channel disconnected
“	E	end of file (EOF) on archive file
“	I	archive request inactivated
“	m	value has a minor alarm status
“	M	value has a major alarm status
“	N	no data for some other reason than listed above

3.1.2 Statistical Data

For statistical data the header contains the same information, namely the process variables' names and units. Again, the timestamps are on the left side. However, each data column has a subcolumn for the mean and another subcolumn for the standard deviation. There is an additional column labeled by "COUNT" which gives the number data points used in the calculation of each mean and standard deviation. Look at Figure 6 for an example.

Notice that when one chooses to retrieve statistical data from the *Data Load Options* window, the text display window automatically displays the statistical data. Also, no data status codes are available.

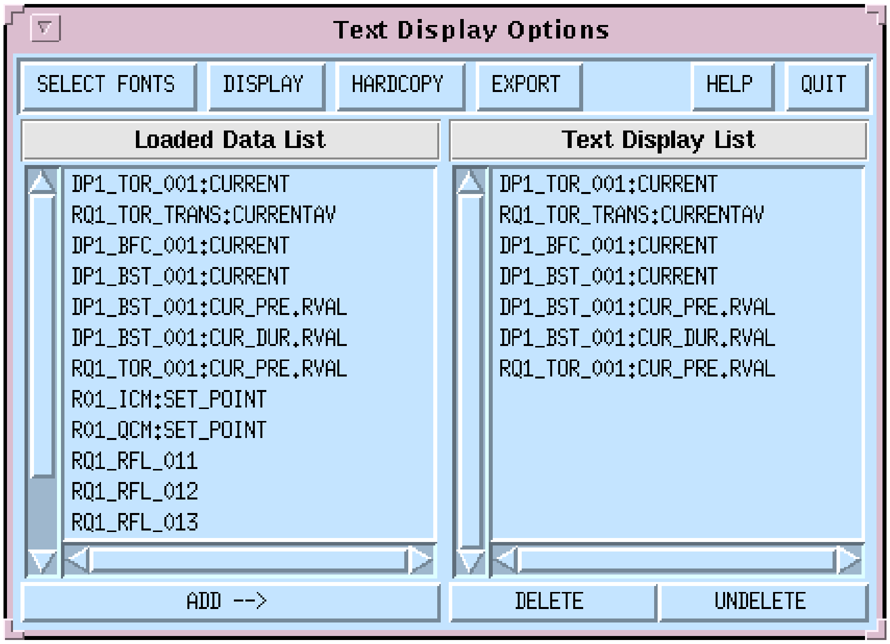
FIGURE 6 text display window for statistical data



3.2 Text Display Options Windows

The display of data can be changed by means of the *Text Display Options* window which one can see by selecting the appropriate *Options* button in the *Process Variable Selection* window. Parameters such as the font, hardcopy and export targets, and list of channels to view can be accessed. See Figure 7 following.

FIGURE 7 text display options window



3.2.1 Lists

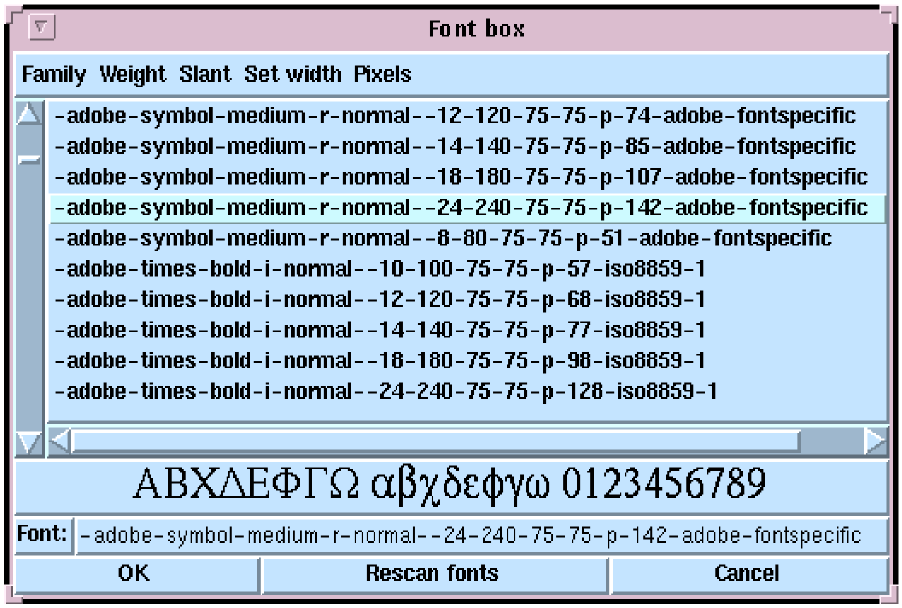
The *Loaded Data List* is identical to the one in the *Process Variable Selection* window. It is present here in order to allow additions to the channels listed in the *Text Display List* next to it by means of the *ADD -->* button. Items from the *Text Display List* are shown in the *Text Display* window. The *DELETE* and *UNDELETE* buttons allow for the manipulation of channels in the *Text Display* window.

3.2.2 Fonts

It is possible to change the format of the text display by selecting different fonts. However, in order to be able to print to a postscript file or printer, it is automatically formatted to fit on a $8^{1}/_{2}$x 11 page. To access the font controls, simply press the *SELECT FONTS* button. This will bring up the *Font box* window seen below in Figure 8. The font box shows sample text of the font chosen so that the user can make judgments about the appropriateness of the selection before making a final choice.

The choice of fonts is taken from a file in the **arr** home directory called **Fonts**. This file can be easily created as it consists of the output of the UNIX command, `xlsfonts`.

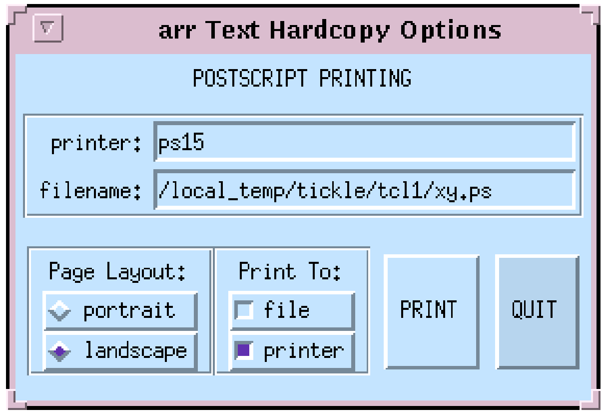
FIGURE 8 font box window



3.3 Hardcopy Window

The hardcopy window allows the user to print to both files and printers in either the portrait or landscape orientation. The printer and/or file names are entered in the appropriate boxes. If no printer name is designated (i.e. it's box is blank) it will automatically use the printer defined in the user's environmental variable "PRINTER". To set this environmental variable, see your computer system administrator. **Unfortunately, the postscript output includes up to one page of the window which is currently being displayed on the computer screen**. This is a limitation of tk version 3.6 and hopefully will be corrected in the future. See Figure 9 below for details.

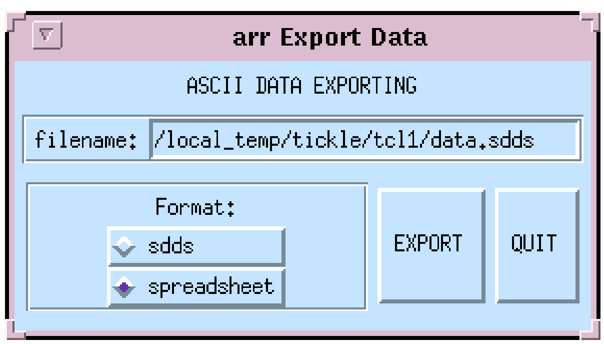
FIGURE 9 text hardcopy options window



3.4 Export Data Window

The data in the text display window can be exported in 2 different ascii formats. The first is the SDDS format originated at Argonne National Lab, and the second is in a tab-delimited spreadsheet format with one line per timestamp. The spreadsheet format is compatible with programs like Excel and Lotus 1-2-3. See Figure 10 below. The user is warned about overwriting existing files.

FIGURE 10 export data window



3.4.1 Spreadsheet Format

Immediately following is a sample output file in the spreadsheet format put into table form (Table 2). The first row is the beginning timestamp. The second and third rows are header information with the second row listing the type of data contained in the data column underneath. Listed in the third row are units if appropriate. If there are no units for a channel, the word “none” will appear in place of the units. The remaining rows are data with the first column being the timestamp. In the second column is the time in seconds referenced from the beginning timestamp. The third column is the data status codes for the first channel, and the fourth column is the data for the first channel. After that, the data status code and data columns are repeated for the other channels.

Part of the reason this format was chosen is because it is largely compatible with that of the older version of the archive retrieval program, **ARR**. There are a few small differences in the second and third header lines. The entry “status” is “stat” and the entry “timestamp” is “date” in the old format. In the third row, “timestamp” and “status” entries have no units whereas “date” had units of “stamp” and “stat” had units of “stat” in the old format.

TABLE 2 ascii output, regular data - spreadsheet format

03/23/93 07:44:23.000					
timestamp	time	status	Pv_#1	status	Pv_#2
	seconds		ma		none
03/23/93 07:44:23.200	0.200000	O	-0.0031	O	10.5720
03/23/93 07:44:23.400	0.400000	O	0.0031	O	11.3043
03/23/93 07:44:23.600	0.600000	O	-0.0031	O	11.7605
03/23/93 07:44:23.800	0.800000	O	0.0092	O	13.0808
03/23/93 09:42:56.600	7113.60	S	22.0829	S	43.8876

The format for statistical data is slightly different than that for regular data. Below in Table 3 the second header line contains a single column of “sample” with the units of “counts” to indicate the number of samples in each mean/stdDev calculation. After “sample” comes pairs of channel names and channel units. In the third header row, changes include the units of “counts” for the “sample” category and pairs of “mean” and “stdDev” under the channel name and channel units respectively.

Again, this format is largely compatible with the output of **ARR**. The exceptions are again that “date” has become “timestamp”, “count” has become “counts”, and the “timestamp” category has no units whereas “date” had the units of “stamp” in the old format.

TABLE 3 ascii output, statistical data - spreadsheet format

03/23/93 07:44:23.000						
timestamp	time	sample	Pv_#1	ma	Pv_#2	%
	seconds	counts	mean	stdDev	mean	stdDev
03/23/93 07:44:23.200	0.200000	15	-0.0	0.0	12.4	1.1
03/23/93 09:42:56.600	7113.600	15	21.9	0.2	43.8	0.0
03/23/93 10:26:32.600	9729.600	15	16.0	0.0	36.6	1.1
03/23/93 10:45:42.400	1.08794e4	15	27.2	0.9	51.6	0.2

3.4.2 SDDS Format

For a more complete reference to the SDDS data format, look at the "Application Programmer's Guide for SDDS Version 1" by Michael Borland at APS (Advance Photon Source). Following is a sample of the output that **arr** gives:

```
! This file contains data retrieved from an EPICS archived
! data file using the "arr" program for viewing, printing,
! or exporting such data. See the "arr Programmer's Guide"
! for more information.
!
SDDS1
&description
   text = Data from archive file: /home/obiwan/gtacs/apple/ar/arSet/bug1.txt
   contents = Archive data
&end
!
&column
   type = string
   name = timestamp
   description = date and time of data
   format_string = %21s
   field_length = 21
&end
!
&column
   type = double
   name = time
   units = seconds
   description = time of data referenced to first timestamp
   format_string = %lg
   field_length = 0
&end
!
&column
   type = character
   name = status1
   description = d01_ttd_006 status code #1
   format_string = %c
   field_length = 1
&end
!
&column
   type = character
   name = status2
   description = d01_ttd_006 status code #2
   format_string = %c
   field_length = 1
&end
!
&column
   type = float
   name = d01_ttd_006
   units = none
   description = d01_ttd_006 value
   format_string = %g
   field_length = 0
&end
!
&column
   type = character
   name = status1
   description = ims_ttd_002:bun1_bot_temp status code #1
```

```
    format_string = %c
    field_length = 1
&end
!
&column
    type = character
    name = status2
    description = ims_ttd_002:bun1_bot_temp status code #2
    format_string = %c
    field_length = 1
&end
!
&column
    type = float
    name = ims_ttd_002:bun1_bot_temp
    units = none
    description = ims_ttd_002:bun1_bot_temp value
    format_string = %g
    field_length = 0
&end
!
&column
    type = character
    name = status1
    description = RQ1_RFL_008 status code #1
    format_string = %c
    field_length = 1
&end
!
&column
    type = character
    name = status2
    description = RQ1_RFL_008 status code #2
    format_string = %c
    field_length = 1
&end
!
&column
    type = float
    name = RQ1_RFL_008
    units = none
    description = RQ1_RFL_008 value
    format_string = %g
    field_length = 0
&end
!
&data
    mode = ascii
    lines_per_row = 1
    no_row_counts = 0
    additional_header_lines = 0
&end
!
! Tabular-Data Columns
!
11/08/92 00:06:14.000   0.000e+00   O   31.90   O   26.20   M   0
11/08/92 00:16:14.000   600.00000   O   31.90   O   26.30   M   0
11/08/92 00:26:14.000   1200.0000   O   31.90   O   26.20   M   0
11/08/92 00:36:15.000   1801.0000   O   32.00   O   26.30   M   0
11/08/92 00:46:16.000   2402.0000   O   31.90   O   26.30   R   0.0391
11/08/92 00:56:16.000   3002.0000   O   31.90   O   26.20   O   0.0391
11/08/92 01:06:16.000   3602.0000   O   32.00   O   26.30   O   0.0391
11/08/92 01:16:14.000   4200.0000   O   31.90   O   26.30   M   0
11/08/92 01:26:14.000   4800.0000   O   31.90   O   26.30   M   0
11/08/92 01:36:14.000   5400.0000   M   0       M   0       M   0
11/08/92 01:46:14.000   6000.0000   R   31.90   R   26.30   M   0
```

The data format in the SDDS case is identical to the spreadsheet format. In other words, all the columns are defined in the same way. Columns are even separated by tabs in both cases.

One important detail to note is that data is stored in the so-called “tabular-data column” format. In this particular format, it is NOT permissible to skip data - to leave a blank space where data should go. **Since there are occasions when data is missing, these are accommodated by substituting a single zero, “0”, for the missing data point and setting the first data status code to “M”** (which means that the data is missing). Thus, the user of SDDS output files from **arr** needs to be aware that these zeros need to be discarded.

CHAPTER 4 Graphical Viewing Of Data

This chapter leads the reader through the means of viewing data in graphical form and printing those graphs to printers and files.

4.1 Graphics Display Options Window

The *Graphics Display Options* window is one used to create, configure, and export all graphs. See Figure 11 for a view of this window and all of its associated menus.

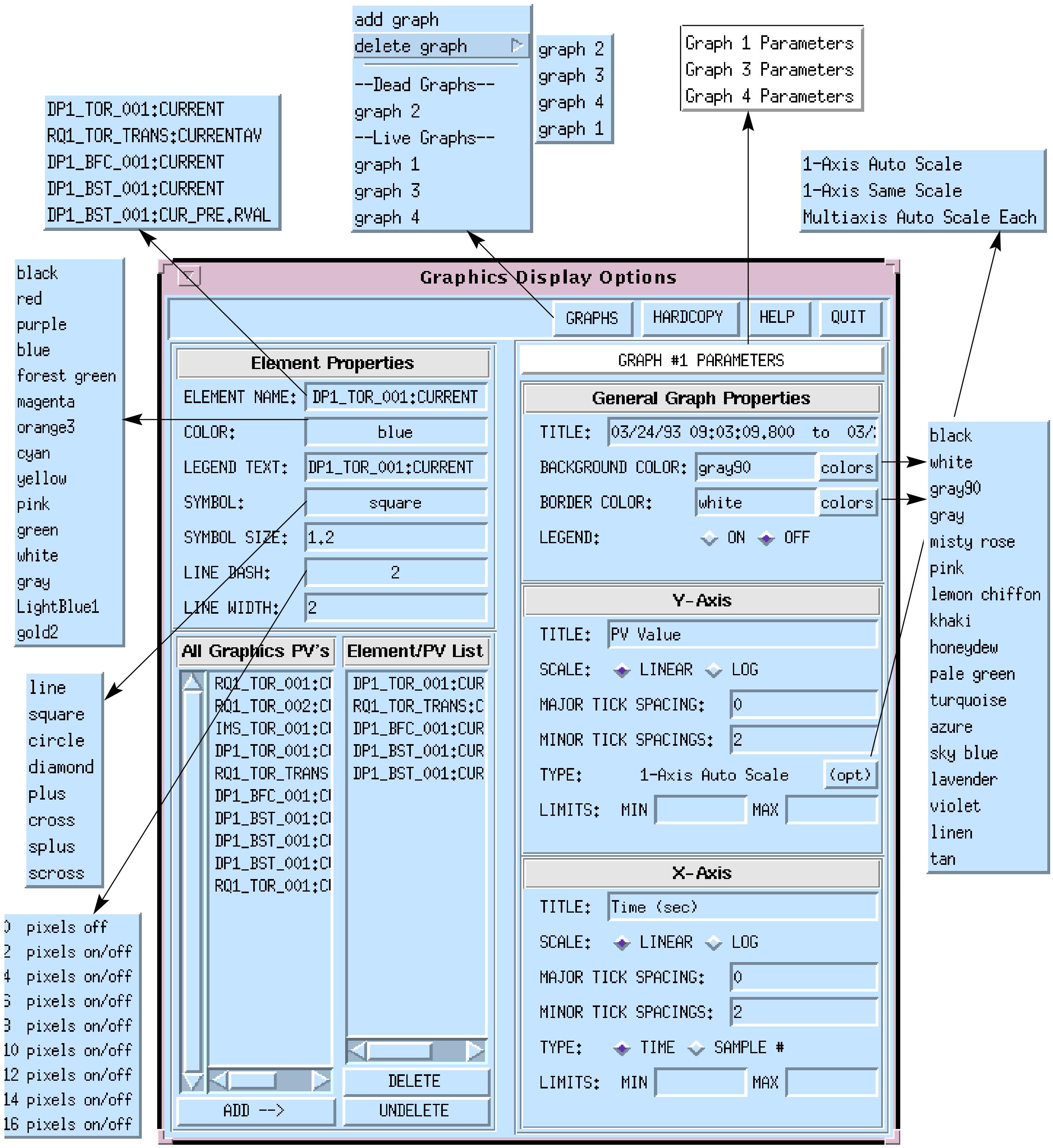
4.1.1 "GRAPHS" Menu

At the top of Figure 11, the *GRAPHS* menu is used to create or delete graphs by selecting either the *add graph* or *delete graph* option respectively. Listed underneath these 2 possibilities are other menu items such as *graph N*. These items simply map the window associated with graph N (if iconified or withdrawn), raise it to the front, load all of graph N's parameters, and update the *GRAPH PARAMETERS* menubutton. Graphs listed under the *Dead Graphs* heading are those which contain data no longer available to **arr** and are not modifiable (for example if a new file is loaded with process variables that are different from those in the graph). While graphs listed under the *Live Graphs* heading are those which can still be modified.

4.1.2 "GRAPH PARAMETERS" Menu

Directly under the *GRAPHS* menubutton is the *GRAPH PARAMETERS* menubutton which tells the user which graph's attributes are loaded into the graphics display options window at the moment. Only that graph can have its's attributes changed until another graph's parameters are selected with that menu. Selecting a graph's parameters with this menu will also have the effect of mapping the graph of interest onto the screen and raising it to the front.

FIGURE 11 graphics display options window and menus



4.1.3 Selecting Process Variables

4.1.3.1 *All Graphics PV's* List

Upon selecting or creating a graph, the next step is to choose which channels one wants to plot. This is done by selecting the channels of interest in the *All Graphics PV's* list and pressing the *ADD -->* button underneath. This will automatically plot the choice in the graph indicated by the *GRAPH PARAMETERS* menubutton. The resulting list of channels can then be seen in the *Element/PV List*. Channels can be added at any time but only up to a maximum of eight. Any further additions are ignored.

Once in the graph, the process variable's data is referred to as a graph "element". Elements names are the same as channel names. Default colors and symbols are assigned when first loading data into a graph but can be easily changed by the user.

4.1.3.2 Element/PV List

The *Element/PV List* is a listing of all the channels in a graph. This can be changed at any time by adding other channels as discussed above. Alternatively, channels can be deleted by selecting those scheduled for termination in the *Element/PV List* and pressing the *DELETE* button directly underneath. Elements that have been deleted can be undeleted with the *UNDELETE* button. Each graph keeps track of its own deletions so that any undeletions are specific to the graph being modified and do not refer to the last item which was deleted if from another graph.

4.1.4 Changing Element Properties

The gray label of *Element Properties* indicates the section of the graphics display options window in which all of an element's attributes can be changed.

4.1.4.1 Element Name

To change the property of an element, the first step is to select the name of the element with the *ELEMENT NAME* menu. The settings of that particular element are then loaded into the rest of the element entries.

4.1.4.2 Color

There are 15 colors in the menu choice. Some of these colors were chosen because of their good visibility against a light background, and some were chosen because of their good visibility against a dark background. See part II of this manual, "Archive Retrieval Tool: arr Programmer's Guide", for desired changes to the color choice.

4.1.4.3 Legend Text

The text of an element's legend entry defaults to the element name. However, any text may be entered for the legend entry.

4.1.4.4 Symbol

Of the 8 types of symbols to choose from, a word of explanation about the last two.The splus and the scross are simply the plus and cross symbols plotted in the background color of the graph. The symbol choices are determined by the BLT graph widget extension of tk.

4.1.4.5 Symbol Size

An element's symbol size can be varied from 0.2 (a couple pixels) to 5 (huge). If *line* is chosen as the symbol, the symbol size is irrelevant.

4.1.4.6 Line Dash

For lines that are drawn, a line dash setting of *0* refers to a solid line. A line dash setting of *N* means that at every *N* pixels of linear distance, the line alternates between foreground and background colors - creating a dashed line.

4.1.4.7 Line Width

The width of a line can vary between 0 and 10. For all symbols except *line*, a width of 0 means that no line is drawn. A line width of 0 for the *line* symbol is the same as a width of 1 - the thinnest line possible.

4.1.5 General Graph Properties

The gray label of *General Graph Properties* indicates the section of the graphics display options window in which all of these attributes can be changed.

4.1.5.1 Title

Whenever a new file is selected, **arr** checks each graph to see if its elements are channels that exist in that file. If these channels exist, their data is loaded and the graphs are updated automatically. In the updating process, **arr** sets the title of each graph to be of the form: "beginning_timestamp to ending_timestamp". This title can be changed at any time to any text.

4.1.5.2 Background Color

The background color of a graph can be changed to either a user-entered value or one of the colors selected from the *colors* menu. In the case of user-entered values, valid symbolic color names such as "black" or "white" are defined in a file called **rgb.txt** in the X library directory (/usr/share/local/x/lib/ in the author's case). Colors can also be specified numerically in terms of their red, green, and blue constituents. There are four forms to choose from as seen in the following table.

TABLE 4 numerical color formats

Format	Color Accuracy
#RGB	4 bits
#RRGGBB	8 bits
#RRRGGGBBB	12 bits
#RRRRGGGGBBBB	16 bits

Each "R", "G", or "B" is one hexadecimal digit of red, green, or blue intensity, respectively. Most workstations have 256 colors (8 bits) available, in which case the #RRGGBB form is adequate. To give some examples, white is #ffffff, black is #000000, and blue is #0000ff.

If **arr** is being run with a monochrome display and a color other than white or black was specified, tk will use white or black depending on the overall intensity of the color that was requested. If **arr** is being run with a color display and all the entries of the color map are used, tk will treat the display as if it were monochrome.

4.1.5.3 Border Color

The background color around the border of the graph can be specified in the same manner as the background color of the actual plotting region (see above).

4.1.5.4 Legend

The legend of a graph can be turned on or off.

4.1.6 Y-Axis Properties

4.1.6.1 Title

The y-axis title defaults to “PV Value” when there is only a single axis (either auto scale or same scale). This can be changed by the user at any time. However, for the multi-axis scale, each y-axis is labeled by its element name only.

4.1.6.2 Scale

Both logarithmic and linear scales are available with the usual restrictions on the log scale. Because the graph widget doesn’t handle the log scale very elegantly, it’s use is somewhat limited.

4.1.6.3 Major Tick Spacing

The major tick marks (ticks labeled by a numerical value) must be specified in terms of the units being plotted. In the case of multi-axis graphs, only the first (inner most) axis can be changed in this way. A default value of 0 allows the graph to decide for itself what labelling it desires.

4.1.6.4 Minor Tick Spacing

The number of spaces between major tick marks can be specified. It defaults to 2. In the case of multi-axis graphs, only the first (inner most) axis can be changed in this way.

4.1.6.5 Type

There are 3 types of y-axes available: 1-axis auto scale, 1-axis same scale, and multi-axis auto scale.

Auto Scale: Auto scaling simply spreads each channels’s data over the whole graph. In both the 1-axis and multi-axis auto scale cases, the data of all the elements are normalized to be identical to the range of the first element in the *Element/PV List*. Elements whose data are constant (flat) are normalized to the middle of the range. If the first element in the list is flat, the graph widget automatically sets a range of “value*1.1” for the maximum limit and ”value*0.9” for the minimum limit. And if the first element contains only zeros, .104 is the max and -.104 is the min.

Same Scale: This type of y-axis puts everything on the same, single scale. When chosen, the entry boxes allowing for manual input of the graph’s maximum and minimum values are activated. The defaults are no values (blanks) which allows the graph to decide for itself what limits it desires.

4.1.7 X-Axis Properties

4.1.7.1 Title

The default title is “Time (sec)”, but can be changed at any time. However, when the x-axis type is switched to *SAMPLE #*, the title automatically changes to ”Sample Number”. Likewise, if the x-axis type should be switched to *TIME*, the title automatically changes to “Time (sec)”.

4.1.7.2 Scale

Both logarithmic and linear scales are available with the usual restrictions on the log scale. Because the graph widget doesn’t handle the log scale very elegantly, it’s use is somewhat limited.

4.1.7.3 Major Tick Spacing

The major tick marks (ticks labeled by a numerical value) must be specified in terms of the units being plotted. A default value of 0 allows the graph to decide for itself what labelling it desires.

4.1.7.4 Minor Tick Spacing

The number of spaces between major tick marks can be specified. It defaults to 2.

4.1.7.5 Type

There are two type of x-axes. The first type is simply time in seconds. The second is sample number or data number which is another way of saying that all data points are plotted equidistantly along the axis. This can be useful if data points are separated by widely varying amounts of time.

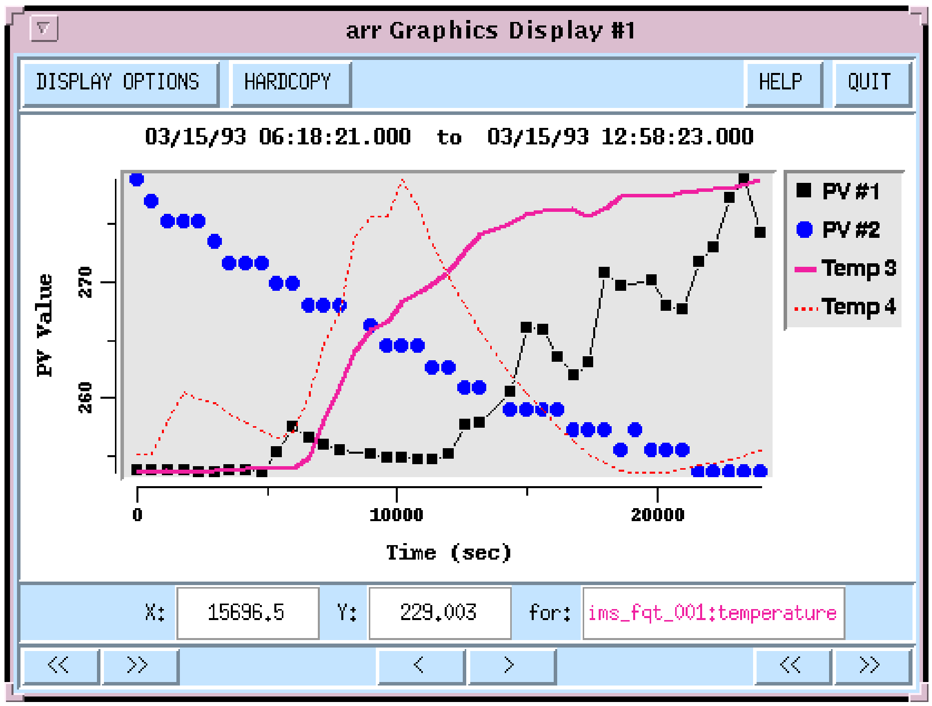
4.1.7.6 Limits

The limits of the x-axis are manually adjustable by entering the user’s values in the min and max entry boxes. The defaults are no values (blanks) which allows the graph to decide for itself what limits it desires.

4.2 Graphics Display Windows (Graphs)

Each graph is a top level window, and the user may create as many of them as desired. There are three main ways by which the user can display graphs on the screen. The *DISPLAY GRAPHIC DATA* button in the *arr Process Variable Selection* window maps all the graph windows onto the screen. Similarly, the *GRAPHS* and *GRAPH PARAMETERS* menubuttons allow graphs to be mapped onto the screen. See Figure 12 below as an example of a graph.

FIGURE 12 graph window (data courtesy of GTA)



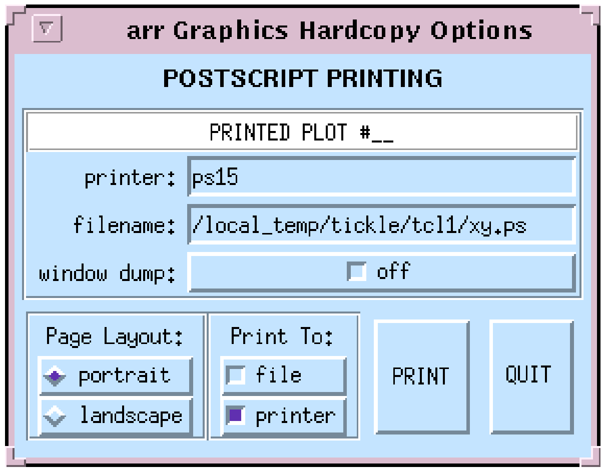
4.2.1 Configuration

Graphs can be configured by means of the graphics display options window as seen in Figure 11. The button labeled *DISPLAY OPTIONS* at the top of each graph will call up this widow. Please refer to section 4.1 for more details.

4.2.2 Hardcopy Window

The hardcopy widow is almost identical to the one used for text data but includes the capability to use window dumps (see Figure 13). With the *window dump* button showing “off”, the *PRINT* button is a menu and allows the user a choice of which graph to print. **In this mode there is a problem with the postscript output for a multi-axis graph. Only the first axis is printed.** The reason is that each of the additional axes are actually additional graph widgets (only partially showing). Since each graph widget has a separate postscript output, there is no way to get multiple axes on one output. However, with the *window dump* button showing “on”, the *PRINT* button will allow one to dump the selected window (click the left mouse button in the window of interest and not in the accompanying message window) to either a postscript file or printer.

FIGURE 13 graphics hardcopy options window



4.2.3 Zoom Feature (mouse)

Although this feature is not readily apparent by looking at the graph, it is possible to zoom in on sections of the graph by use of the mouse buttons. By pressing down on the left mouse button, moving the mouse to another spot (keeping the button pressed), and then releasing, the user can rubberband an area to be expanded. The graph can return to its original settings by clicking the middle mouse button on the graph.

4.2.4 Pan/Zoom Feature (<<, >>, <, > buttons)

The graph can also be zoomed and panned upon by the arrow buttons on the bottom of each graph window. The double arrows buttons on one side of the graph will expand or contract that one edge of the graph by 10% (if possible). Similarly, the single arrow buttons will pan the graph by 10% either to the left or right (if possible). One can, again, undo all the pans and zooms by clicking on the graph with the middle mouse button. If the x-axis is a log scale, these buttons are disabled due to graph widget difficulties.

4.2.5 X, Y Graph Data

Data from each of the channels that are plotted are accessible simply by moving the mouse around the graph. In Figure 12, the numbers following the *X:* and *Y:* labels are the data for the channel shown on the menubutton next to the *for:* label. If the y-axis is set to *1-Axis Same Scale*, then the X,Y data are the same for all channels. If the y-axis is set to either of the auto scales, then the user needs to select the channel of interest with the menu conveniently provided. Due to difficulties inherent in the graph widget, X,Y data is available only for the first process variable in the element list if the y-axis is a log scale.

APPENDIX A Tcl/Tk Conventions

A.1 Tcl/Tk Conventions

To the author, it seems that every Graphical User Interface (or GUI) that I encounter has different conventions for the behavior of entry boxes, menus, and the like. In order to ease the transition to a tcl/tk based GUI, some of the behaviors of these widgets are outlined here. These conventions are the standards supplied with tcl/tk but can be changed by the programmer. Thus, it is possible that not all tcl/tk based programs will behave similarly.

A.1.1 Entry Widgets

An entry widget has the following behavior:

1. Clicking the left mouse button positions the insertion cursor.
2. Dragging with the left mouse button strokes out a selection.
3. The ends of the selection can be adjusted by dragging with the left mouse button down while the shift key is pressed.
4. The view in the entry can be adjusted by dragging with the middle mouse button.
5. Control-h, Backspace, and Delete keys erase the character just before the insertion cursor.
6. Control-w erases the word just before the insertion cursor.
7. Control-u clears the entry to an empty string.
8. Control-v inserts the current selection just before the insertion cursor.
9. Control-d deletes the selected characters.

A.1.2 Menus

A menu has the following behavior:

1. Pressing and releasing the left mouse button within the menubutton will keep the menu posted until a selection is made.
2. Pressing the middle mouse button with the menubutton posts a “disconnected” or “torn off” menu which can be dragged around on the screen with the middle button. It will not unpost when entries are selected. To close a torn off menu, click the left mouse button over the associated menubutton.

A.1.3 Listboxes

Listbox widgets have the following behavior:

1. Clicking with the left mouse button over an item will select that item.
2. Dragging with the left mouse button depressed will select a range of items.
3. The ends of the selection can be adjusted by dragging with the left mouse button depressed while the shift key is down.
4. Selected items can be deselected by clicking or dragging with the right mouse button.
5. Disjointed selections (does not undo the previous selection) can be make by depressing the control button and clicking or dragging the left mouse button.