Array Display Tool ADT

Reference Manual

Kenneth Evans, Jr.

Version 1.2

December 1995

Contents

- Overview
- New Features
- Getting Started
- Display Areas
- Zoom Area
- Mouse Operations
- File Menu
- Options Menu
- View Menu
- Clear Menu
- Installation
- Configuration
- PV Files
- Lattice Files
- Snapshot Files
- Reference Files and Referencing
- Application-Defaults File
- Status Indicators
- Color Code
- Xorbit Simulation Mode

Overview

Array Display Tool (ADT) is a Motif[1] program to display arrays of process variables from the Advanced Photon Source control system. A typical use is to display the horizontal and vertical monitor readings.

The screen layout, apart from the menu bar, consists of two types of graphic areas in which the values for the arrays of process variables are shown: Display areas, which display one or more arrays as a function of index, and a zoom area. In the zoom area specified arrays only are displayed as a function of lattice position along with symbols for the major elements of the lattice. There can be several display areas, but at most one zoom area. When the screen is resized these areas change size proportionally.

There are a number of options in the View Menu to change the way the values are displayed. It is also possible via the Options Menu to:

- Store the current values internally.
- Store the values from a snapshot file internally.
- Display one of the stored sets of values along with the current values.
- Display the difference of the current values with one of the stored sets of values.
- Write the current values to a snapshot file.

There are several (currently 5) slots in which you can store values internally. In addition you can display the values with specified reference values subtracted.

New values are sent to the program anytime the process variables change outside of their dead band (as specified in the control system record for that process variable). If the dead band is chosen appropriately, this should result in less traffic over the control network than if all of the values were polled

[1]Motif is a standard graphical interface for X Windows systems. It is characterized by a three-dimensional look and feel.

at fixed intervals. The current values that have been received are displayed whenever the screen updates. It is also possible to manually update the variables via the Options/Epics/Rescan menu.

The program continuously updates and displays the standard deviation, average, and maximum absolute values for each array and will show the envelope of recent values if desired. The time between updates can be changed via the View Menu or may be specified in the PV file. Whether the statistics are for the current values or accumulated over a period of time is controlled by the Accumulated Statistics button in the Options Menu.

There is a configuration file that is needed to get ADT running. Most of the features of the screen layout, including what process variables are shown in what display areas, are determined by a PV file. The contents of this file determine such things as curve color, units, and scaling. Another relevant file is a lattice file, which specifies the symbols and longitudinal lattice positions of the elements in the lattice. Sets of values can be saved and restored via snapshot files.

Most of the available options can be discovered intuitively by looking through the menus. This manual is primary for reference and is not a tutorial.

New Features

The following features are new in ADT, compared to its predecessor, Orbitscreen.

- There may be an arbitrary number of display areas, practically about 1-4.
- More than one array may be displayed in a display area, making comparisons easier.
- The zoom area and which arrays appear in the zoom area are arbitrary.
- All the display areas scale proportionally when the screen is resized.
- The scale button is replaced by a units per division button, more like an oscilloscope.
- The center value of a display area may be specified.
- A file of reference values (such as BPM offsets) may be specified and turned on/off.
- Clicking the right button in a display area clears the Max/Min envelope display.
- An array may be displayed with a log scale (appropriate for vacuum readings, for example).
- ADT may be started from the command line with a specified PV file.
- The home directory may be specified on the command line.
- Most of the view options may be specified in the PV file.
- Snapshot files retain the paging by array so they are more easily used with SDDS tools.
- The Load menu has submenus.
- The statistics (SDEV, AVG, and MAX) can be accumulated over the maximin/minimum history period. See Average Statistics under the Options Menu.
- The application defaults file, ControlApp, is no longer needed.

Getting Started

In order for ADT to run properly, it needs to find the **adtrc** file. (It is possible to run ADT without the **adtrc** file, but it is not suggested.) This file contains the File/Load menu options and the locations of other files that ADT needs. More about the **adtrc** file may be found under configuration.

The command line for ADT is of the form:

adt [**Standard X O**ptions] [-x] [-f *pvfilename*] **[**-a *adthome***]**

It accepts any of the standard X-Window options, such as **-display machine:0**. The other options are:

-**a** *adthome*	Specify the directory for the configuration file to be *adthome*.
-**f** *pvfilename*	Load the PV file, *pvfilename*, at startup, bypassing the Load menu.
-x	Use Xorbit Simulation Mode.

Both *adthome* and *pvfile* must include a path if they are not in the directory from which adt is started.

Display Areas

In the display areas the values of the process variables are displayed in graphical form with uniform horizontal spacing between the points. Display options, such as whether the values are displayed as a line plot, as an impulse plot, or as markers only, are determined by settings that can be made in the View menu. There are mouse operations for displaying the numerical values of the three elements closest to the mouse cursor and moving the zoom window so the element closest to the mouse cursor is in the center, as well as for clearing the Max/Min envelope (the region between the maximum and minimum values achieved for each process variable since the last reset). There are options in the View menu to show or not show the Max/Min envelope. The Max/Min envelope can also be cleared from the Options menu. Clearing the Max/Min envelope also clears the accumulated statistics.

The units per division of the display area can be changed by a control above it, as can the center value. There are also arrow buttons to change the center value by one unit per division. The starting units per division, center values, and a factor by which to multiply the raw data to get the desired units are set in the PV file, which also determines which process variables are displayed and the heading and units.

The standard deviation, average, and peak absolute values are shown and updated when the values are updated. (The standard deviation defined with n rather than n-1 is used since the whole population is known.) The peak absolute value includes the sign of the value that was largest. The update time interval can be changed in the View/Timing menu. If a log scale is used, the values displayed are the logirithms (base 10) of the actual values.

Zoom Area

Values for arrays are shown in their true relative lattice positions in the zoom window, along with symbols representing the lattice elements. Only those arrays so designated in the PV file are shown in the zoom window. The color for each array is the same as is used in its display area and may be specified in the PV file. There are mouse operations that scroll the zoom window. The zoom area can also be scrolled to the position of a desired element by clicking on that element in the display areas.

The units per division and the center value of the zoom window can be changed by the controls above it the same as for the display areas. The default units per division is the largest of those specified in the PV file and the default center value is the average. The length of the interval of the lattice shown in the zoom area an be changed via the box labeled **Interval:**. The sector shown in the box labeled **Sector:** is the sector that is at the center of the display area and can be varied by typing a new number or using the arrow buttons. In addition, numbers designating the sectors and a mark identifying the center of the display area are displayed in the zoom area to help identify the sectors and the midpoint for mouse scrolling.

The zoom area can be turned off entirely, and it does not appear if the process variables cannot be matched with the lattice elements in the lattice file or if no arrays are specified to appear in the zoom area.

Mouse Operations

In the display areas:

Button 1: Pushing and holding Button 1 brings up a dialog box with the numerical values of the three elements closest to the mouse cursor. There is an arrow next to the one that is closest to where the mouse was clicked. The dialog box goes away when you release the mouse button within the display area. If you want the box to stay up, drag the mouse cursor off the display area before releasing it.

Button 2: Pushing Button 2 moves the element closest to the mouse cursor to the center of the zoom area. If there is more than one array in the display area, the element corresponding to the first array in that display area is used.

Button 3: Pushing Button 3 in any display area resets the Max/Min envelope in all areas.

In the zoom area:

Button 2: Pushing Button 2 moves the point under the mouse cursor to the center of the zoom area. (Similarly to what Button 2 does in the upper two display areas.)

Button 3: Pushing and holding Button 3 scrolls the zoom area horizontally. The scroll speed is proportional to the distance of the mouse cursor from the center of the zoom area. You can scroll either direction, depending on which side of the center you hold the mouse cursor.

File Menu

Load

The Load button brings up a menu with submenus of predefined screen layouts that may be loaded into the display. Each screen layout corresponds to a PV file and the associated lattice file. As soon as the screen layout is loaded, monitoring starts. The screen layouts on the menu should be the most commonly used ones. If you wish to specify another layout, you may pick the menu item labeled **Custom**. In that case, a file selection dialog box to pick the PV file name will appear. The items that appear in the menu are customizable and are specified in the configuration file. A specified PV file may also be loaded via the command line.

Read Reference

The Read Reference button brings up a file selection dialog box to allow you to pick a reference snapshot file containing reference data.

Read Snapshot

The Read Snapshot button brings up a menu of possible slots in which to store data from a snapshot file. After choosing the slot from the menu, there will be a file selection dialog box to allow you to pick the snapshot file containing the data. The data stored in the slot can then be used for displaying or differencing with the current data. The snapshot file must contain the same arrays with the same elements as the PV file specifying the current screen layout, or it will be rejected.

Write

The Write button allows you to store either the current data or the data saved in one of the slots in a snapshot file. It brings up a menu which includes the current data and the possible slots. After choosing the current data or the desired slot, there will be a file selection dialog box to allow you to pick the snapshot file into which to write the data.

Plot

The Plot button brings up an SDDS plot of either the current data or the data saved in one of the slots.

Status

The Status button brings up a dialog box with information about the screen layout, the accumulated statistics, and the data loaded into slots.

Quit

The Quit button closes any Channel Access connection and terminates ADT.

Options Menu

EPICS

Normally, using File/Load is sufficient to perform the required EPICS procedures. This item provides more control over EPICS. The **Start** button will initialize EPICS if it is not initialized and start it with the current PV file. If EPICS is stopped, the **Start** button will start it and rescan the process variables. Initialized means connections have been established with all the process variables. Started means it is initialized, is receiving messages whenever those values change outside of their dead bands, and is updating the display. When it is started, the **Start** button changes to a **Stop** button. Consequently, this button also tells you what state it is in. The **Stop** button leaves EPICS initialized but not receiving messages and not updating the display. You can play with the existing data. The **Exit** button stops EPICS and closes the connections, so it is no longer initialized. The **Reinitialize** button is equivalent to **Exit** followed by **Start**. The **Rescan** button causes EPICS to explicitly update all the values whether they have changed out of the dead band or not. It insures the readings are current.

Store

The Store button allows you to store the current values in one of the available slots. The status of what is stored in the slots can be displayed with the File/Status button.

Display

The Display button allows you to display the current values from one of the available slots in addition to the current values. They will be drawn in the stored data color. This capability allows you to compare the current values, which are being updated, with some other set of values. **Off** turns off the display of these other values. The status of what is stored in the slots can be displayed with the File/Status button.

Difference

The Difference button allows you to display the difference of the current values and the values in one of the available slots. This capability allows you to see the changes in the current values from some other set of values. **Off** turns off the differencing. The center value is reset to zero for all display areas when differencing is on and reset to the former values when it is turned off. The status of what is stored in the slots can be displayed with the File/Status button.

Check Status

The Check Status button allows you to choose how much status information is displayed. The choices are **Off** (do not display status symbols), **Check InValid** (display symbols only for invalid readings), and **Check All** (display all status symbols). ADT still receives all status information regardless of these choices. The choices only determine how much of the information is displayed. The button is only operational if status process variables have been specified in the PV file.

Reference Enabled

The Reference Enabled button turns referencing on and off. Turning it off will save processing time. Referencing will be off in any event, unless a reference file has been specified in the PV file or via the File menu. The reference enabled switch is retained from screen layout to screen layout, so if referencing is disabled for one layout, it will remain disabled for succeeding layouts.

Zoom Enabled

The Zoom Enabled button turns the zoom window on and off. Turning it off will save processing time. There will be no zoom window in any event, regardless of the setting of this button, if the process variables cannot be matched with the lattice elements. The zoom enabled switch is retained from screen layout to screen layout, so if zoom is disabled for one layout, it will remain disabled for succeeding layouts.

Accumulated Statistics

The Average Statistics button toggles the statistics displays (SDEV, AVG, and MAX) between the current values and those averaged over the Max/Min history period for SDEV and AVG or the maximum absolute values during the history period for MAX. The averaging is reset when the history is reset. See the next item.

Reset Max/Min

The Reset Max/Min button resets the stored maximum and minimum values for each process variable to the current values. This effectively restarts the Max/Min history. The display of the Max/Min envelope is controlled by the Max/Min and Filled Max/Min options in the View menu. The Max/Min envelope can also be reset by clicking with the third mouse button in any display area.

View Menu

Defaults for all of the View Menu items may be specified in the PV file. They are reset to the built-in defaults when the screen layout is changed.

Timing

The Timing button brings up a dialog box that lets you change the screen update interval. Recall that new values are received by ADT when the process variables go out of their dead band. These values are collected and are displayed only when the screen updates.

Markers

The Markers toggle button toggles whether markers are shown or not for the data points in the upper two display areas.

Lines

The Lines toggle button toggles whether lines are shown or not for the data points in the upper two display areas. If Bars is chosen, then Lines will not be, however.

Bars

The Bars toggle button toggles whether impulse lines are shown or not for the data points in the upper two display areas. Bars overrides Lines.

Grid

The Grid toggle button toggles whether grid lines are shown or not in the upper two display areas.

Max/Min

The Max/Min toggle button toggles whether the Max/Min envelope is shown or not in the display areas. The Max/Min envelope is reset with the Options/Reset Max/Min button or clicking the third mouse button in any display area.

Filled Max/Min

The Filled Max/Min toggle button toggles whether or not the area between the maximum and minimum values is filled with gray. Filling the area makes it easier to see where the values have been during the history period. It does nothing if the Max/Min button is not set on.

Clear Menu

Clear

The Clear button clears all three display areas. Unless EPICS is stopped, they will be redrawn at the next scheduled update.

Redraw

The Redraw button redraws all three display areas with the last values used, not the current ones. Unless EPICS is stopped, they will be redrawn at the next scheduled update anyway.

Update

The Update button redraws all three display areas with the current values. Unless EPICS is stopped, they will be redrawn at the next scheduled update anyway.

Autoclear

The Autoclear button toggles whether new values erase old ones or not. Turning Autoclear off is another way to obtain a history of the values.

Installation

In the typical, standard installation there should be a directory in which most of the ADT files are found. The files that should be in this directory are:

adtrc
adt.html
aps.icon

There should be subdirectories **pv** for PV and lattice files and **snap** for snapshot files. There may be a similar set of subdirectories, **xorbit/pv** and **xorbit/snap**, for files used with Xorbit simulation.

All of these files may be located elsewhere if desired. See Configuration.

The help package also refers to screen layout files:

pv/par.bpm.pv
pv/par.lat
snap/par.bpm.ref.snap

that are assumed to be in subdirectories, **pv** and **snap**, relative to the location of **adt.html**. If they are not there, you will not be able to view them from Mosaic.

The executable file, **adt**, should be located in a directory somewhere in your **$PATH**.

The application defaults file, **ControlApp**, is not required or necessary. If it is used, it should be located as specified in the section on the application defaults file. You can specify personal modifications to the resources in your **.Xdefaults** (or equivalent) file or in your own **ControlApp** file in your home directory. (See an X Windows manual for the involved search path that X uses to find resources.)

Configuration

Exactly what is displayed in ADT is defined via four types of files. There are PV files, lattice files, and snapshot files, which are described below. Reference files are also snapshot files. The fourth file is a configuration file, described here, that defines what appears in the File/Load menu as well as where the default PV files, lattice files, snapshot files, and the help files are located.

The directory for the configuration file may be specified on the command line with the -a option, in which case the program looks for a file named **adtrc** in that directory. If not specified, the program looks for it first in your home directory with the name **.adtrc** (with a dot). If one is not found, it next looks in the directory given in **$ADTHOME** if it exists, otherwise in the current working directory, for a file named **adtrc** (with no dot). If it still has not found one or it is invalid, there will be no options in the File/Load menu except Custom; the PV and snapshot files will be assumed to be in subdirectories, **pv** and **snap**, to the current working directory; and the help files will be assumed to be in the current working directory.

This configuration file is an SDDS[2] file. This manual assumes you are familiar with SDDS files. If not, you may need to consult the SDDS documentation to understand the meaning of terms such as parameter and fixed_value. The configuration file has string parameters, **ADTPVDirectory,** which specify the directory for the PV and lattice files, and **ADTSnapDirectory**, which specifies the suggested directory for the snapshot files. (Independently of these settings, Custom PV, custom lattice, and snapshot files may be read from or written to any directory by changing the filter in the file dialog box.) There are corresponding parameters, **ADTXPVDirectory** and **ADTXSnapDirectory**, which may specify different versions of these directories if Xorbit simulation is used. It is not necessary to do this. If the first two parameters are not specified and **$ADTHOME** exists, then they will default to **$ADTHOME/pv** and **$ADTHOME/snap,** respectively, and will default to **./pv** and **./snap**, respectively, otherwise. If the Xorbit parameters are not given, they will default to whatever is used for **ADTPVDirectory** and **ADTSnapDirectory**.

---

[2]SSDS stands for Self-Describing Data Sets and is a format developed by Michael Borland to provide a flexible and standard, general purpose, file format. There are routines available for reading and writing SDDS files.

There is also a string parameter, **ADTHelpFile**, which specifies the location of the Mosaic[3] HTML help file. This file should be in **$ADTHOME/adt.html**. It defaults to **$ADTHOME/adt.html** if **$ADTHOME** exists and to **./adt.html**, otherwise. If you are reading this in a browser, such as Mosaic, then you are probably browsing a version of the **adt.html** file.

If any file locations are specified through the above parameters, it is suggested that full pathnames be used.

Each page in the configuration file corresponds to one submenu. The required parameter **ADTNMenus** specifies how many submenus (pages) there will be. It is required because there is currently no way to tell how many pages are in an SDDS file until it has been read and the number is needed to allocate space. **ADTNMenus** applies to the whole file and should be a fixed_value, short parameter. The Parameter **ADTMenuTitle** is a string parameter which specifies the label for each submenu button.

There are two string columns, **ADTPVFile** and **ADTMenuLabel**, which give the name of a PV file (without the path) and the label that will appear in the File/Load submenu in order to select this file. These PV files will be expected to be in the **ADTPVDirectory**. If these columns are not there, there will be no options in the File/Load menu except Custom.

An example configuration file should be located in **$ADTHOME/adtrc**.

Parameter Summary

ADTHelpFile, string, fixed_value
ADTMenuTitle, string, **Required**
ADTNMenus short, fixed_value, **Required**
ADTPVDirectory, string, fixed_value
ADTSnapDirectory, string, fixed_value
ADTXPVDirectory, string, fixed_value
ADTXSnapDirectory, string, fixed_value

Column Summary

ADTMenuLabel, string, **Required**
ADTPVFile, string, **Required**

PV Files

The PV files specify what process variables are to be displayed in each of the display areas, and many of the other factors that determine the screen layout. The PV files are valid BURT[4] request files. They are also SDDS files. This manual assumes you are familiar with SDDS files. If not, you may need to consult the SDDS documentation to understand the meaning of terms such as parameter and fixed_value.

You can make new PV files containing any process variables you wish to monitor, and you can save them in your own directories. The easiest way to make a PV file is to find an existing one and

---

[3]This manual was designed to be printed and also viewed on-line in Mosaic. Mosaic is an information browser that connects to many information sites on the Internet and World Wide Web. It was developed at the National Center for Supercomputing Applications at the University of Illinois.
[4]BURT is the backup and restore program for the APS control system.

modify it. It should be fairly obvious where the changes are to be made. These can then be read in from the File/Load/Custom menu. An example PV file should be located in **$ADTHOME/pv/par.bpm.pv**.

There are three types of SDDS parameters that appear in a PV file:

Global Parameters: These refer to the file as a whole and are typically entered as fixed_value quantities.

Array Parameters: These vary from array to array and refer to the array in question.

Area Parameters: These refer to the display area associated with the array rather than the array. If there is only one array per display area, there is no effective difference between these and array parameters. When there are several arrays assigned to one display area, even though the parameter is specified on the page for the array, it really belongs to the display area, and since it can only have one value, the last one specified is used.

The possible parameters that may be specified in a PV file are:

ADTFileType: A required, string parameter that should be "ADTPV". This is a global parameter that merely serves for file checking.

ADTNarrays: A required, short parameter that specifies how many arrays (pages) are in the file. It is required because there is currently no way to tell how many pages are in an SDDS file until it has been read, and it is necessary to allocate space beforehand. This is a global parameter.

ADTNareas: A short parameter that specifies how many display areas are in the file. It is required because there is currently no way to know it until the file has been read, and it is necessary to allocate space. If given, it must be the correct number for the display areas assigned by **ADTDisplayArea**. No unused display areas are allowed. If not given, there will be one array per display area. This is a global parameter.

ADTLatticeFile: A string parameter that specifies the lattice file to use for the zoom area. If not given, there will be no zoom area. This is a global parameter.

ADTReferenceFile: A string parameter that specifies the reference file to use. If not given, there will be no referencing. This is a global parameter.

ADTTimeInterval: A short parameter that specifies the time interval in milliseconds between screen updates. If not specified, the built-in default (currently 3000 ms) will be used. This is a global parameter.

ADTMarkers, ADTLines, ADTBars, ADTGrid, ADTMaxMin, ADTFilledMaxMin: These short parameters specify the default settings for the toggle buttons in the View menu. Positive is true, and 0 is false. If not specified, the built-in defaults will be used. These are global parameters.

ADTHeading: A string parameter that specifies the heading part of the title that appears above the display areas. The title includes the heading followed by the units in parentheses. There is a built-in default.

ADTUnits: A string that specifies the units part of the title that appears above the display areas. The default is a blank string.

ADTColor: A string parameter that specifies the named color to use for this curve. The name must be in the X Windows color database. Case is not important. If not specified, a built-in default will be used. Names can be of the form #rrggbb (like #110000 for Red).

ADTZoomArea: A short parameter that specifies whether this array is to appear in the zoom area. Positive is true, and 0 is false. The default is false.

ADTLogScale: A short parameter that specifies whether this array is to appear as the logrithm (base 10) of the value. Positive is true, and 0 is false. The default is false.

ADTScaleFactor: A double parameter by which all the raw data will be multiplied before being displayed. For example, to convert raw data in millimeters to displayed values in meters, the **ADTScaleFactor** would be 0.001, and the **ADTUnits** would be "m". If this parameter is not supplied, it will be taken as 1.0.

ADTDisplayArea: A short parameter that specifies in which display area to display the array. The display areas are numbered starting with 1 at the top. The default is to display one array per display area.

ADTUnitsPerDiv: A double parameter that specifies the default units per division for the display area for that array. If there is more than one array for that area, the last one is used. If this number is not one of the available values, the next-higher available value will be used. If this parameter is not supplied, the built-in default will be used. This is an area parameter.

ADTCenterVal: A double parameter that specifies the value that corresponds to the horizontal axis. The default is 0.0. This is an area parameter.

The PV file has one required string column, **ControlName**, which contains the names of the process variables in the array. Two other string columns are required for the PV file to be a valid BURT request file, a useful thing to do. These columns are **ControlType** and **ControlMode**. The control type is the string "pv" for process variable and the control mode is usually "RO" for read only. The PV file will have one more column for **StatusName** if status is implemented. In that case the column contains the names of the process variables that specify the status.

Parameter Summary

ADTBars, short, fixed_value
ADTCenterVal, double
ADTColor, string
ADTDisplayArea, short
ADTFileType, string, fixed_value, **Required**
ADTFilledMaxMin, short, fixed_value
ADTGrid, short, fixed_value
ADTHeading, string
ADTLatticeFile, string, fixed_value
ADTLines, short, fixed_value
ADTLogScale, short
ADTMarkers, short, fixed_value
ADTMaxMin, short, fixed_value
ADTNAreas short, fixed_value
ADTNArrays short, fixed_value, **Required**
ADTReferenceFile, string, fixed_value
ADTScaleFactor, double
ADTTimeInterval, short, fixed_value
ADTUnits, string
ADTUnitsPerDiv, double
ADTZoomArea, short

Column Summary

ControlMode, string, **Required** only for BURT compatibility
ControlName, string, **Required**
ControlType, string, **Required** only for BURT compatibility
StatusName, string

Lattice Files

The lattice files contain information about all the basic lattice elements, including their longitudinal coordinates. This information is used to draw the symbols for the magnets in the Zoom Window and to determine the longitudinal coordinates corresponding to the process variables, so that the values can be shown in their true relative locations in the Zoom Window. There should only be one lattice file for each ring so you should have little need to make one yourself. An example lattice file should be located in **$ADTHOME/pv/par.lat**.

The parameters that must be specified in a Lattice file are:

ADTFileType: A required, fixed value of type string that should be "ADTLATTICE".

Nsectors: A required, fixed value of type short that is the number of sectors into which to divide the lattice for the purposes of scrolling.

Stotal: A required, fixed value of type double that is total length of the lattice in meters.

Ring: A required, fixed value of type short that is 1 for a ring and 0 for a beamline.

There is a program, **xintolat**, that converts Xorbit input (*.xin*) files to lattice files. Its usage is:

xintolat [**-h**] [*file1.xin*] [*file2.lat*]

-h Help

If the filenames are not specified, the program will prompt for them.

The columns that must be specified in a lattice file are:

S: A required column of type double that is the position of the start of the element in meters.

Length: A required column of type double that is the length of the element in meters.

SymbolHeight: A required column of type short that is the relative height of the symbol for the element. Correctors should use 1, quadrupoles should use 2, and sextupoles should use 3. The sign of the number should indicate the sign of its strength.

Name: A required column of type string that is the name of the element. ADT will match elements to process variables by looking for this string as a substring of the process variable name.

Parameter Summary

ADTFileType, string, fixed_value, **Required**
Nsectors, short, fixed_value, **Required**
Ring, short, fixed_value, **Required**

Stotal, double, fixed_value, **Required**

Column Summary

S, double, **Required**
Length, double, **Required**
SymbolHeight, short, **Required**
Name, string, **Required**

Snapshot Files

Snapshot files may be saved via the File/Write menu. Depending on the options in the menu, these files could contain the values of the displayed process variables at the current time, at the time the orbit was saved via the Options/Save menu, or at the time given in another snapshot file read via the File/Read menu.

These snapshot files should be equivalent to the ones generated by BURT, except that they are paged the same way as the PV files (which are valid BURT request files). BURT collects all the process variables in all the pages of a request file, alphabetizes them, and puts the results in a single page in the snapshot file. This makes it hard to extract the information that belongs to each array. There is no loss, other than a lack of strict compatibility, in retaining the array pages and not alphabetizing. Either the conventional BURT snapshot files or the ones from ADT are valid request files for BURT. Therefore, they can be used to restore the process variables in them if these process variables are capable of being restored. (Monitor readings and corrector read backs cannot be restored, for example. Corrector set points can.) You can save snapshot files in your own directories. An example snapshot file should be located in **$ADTHOME/snap/par.bpm.snap**.

The snapshot file will have two more columns, **StatusName** and **StatusValue**, if status is implemented, also unlike a BURT snapshot. You cannot restore the status, so this should not be a problem. For definitions of the other parameters and columns, see the BURT documentation. A summary is given here, however.

Parameter Summary

SnapType, string, fixed_value
TimeStamp, string, fixed_value

Column Summary

ControlName, string
ControlType, string
Count, long
Lineage, string
ValueString, string

Reference Files and Referencing

It is possible to read in a file of reference values, either via the PV file or via the File Menu. These values will be subtracted from the raw data before it is displayed. Even if a reference file has been read, referencing can be enabled or disabled via the Options Menu. This feature can be used to specify monitor offsets, for example, and to display the monitor readings relative to the monitor offsets, which is usually a more desirable thing to do. Referencing is similar in operation to differencing, except that it can be specified in the PV file. You can have both differencing and referencing on at the same time. The

reference values in this case, however, will be subtracted from both the data array and the difference data array, which are then subtracted, so it is the same as having referencing off. (The code is smart enough to avoid the subtracting.)

Any snapshot made from the screen layout may be used for a reference file, and the requirements for a reference file are the same as those for a snapshot file. In particular, the reference file must contain the same arrays with the same elements as the PV file specifying the current screen layout. If this snapshot file cannot be made by saving values from ADT, then it would probably be wise to take a file that has been so saved and change the values.

Application-Defaults File

The application-defaults file for ADT is named **ControlApp**. It is not needed in order to run ADT since there are adequate fallback resources defined in the program. The usual place for such files is in **/usr/lib/X11/app-defaults**, and **ControlApp** should already have been put there by a system manager if it is used, so that you do not have to worry about it. You may also have an additional application-defaults file named **ControlApp** in your home directory. You can put your own resource customizations for ADT in this file or in your **.Xdefaults** file. The latter is preferred. Note that all of the resources in your private file will remain in effect even if the program is changed in a way that requires the system **ControlApp** to be changed. Consequently, it is advisable to keep as few resources as possible in your personal version or to not have a personal version at all, especially an out-of-date one. If you do not understand about resources or application-default files, see your local administrator.

If you do need to changes the resources for your personal needs, the resources are listed in resources.h, a file that is included in the distribution for ADT. ADT may also be used with the X client, Editres, to find the resource names, change them, and experiment with them.

Status Indicators

ADT can be made to display the status of BPM readings. The status values must be available in a process variable which can have the enumerated values:

0 = InValid
1 = Valid
2 = OldData

In order to enable this capability there must be an additional string column in the PV file called **StatusName**. Each entry in this column should contain the name of the process variable that has the status information for the process variable that is to be displayed. If there is no status name or you don't want to use it, the entry should be a "-". If the column is not there, there will be no indication of status other than for process variables that are not connected.

When this has been done, the display should be the same as usual when the data is **Valid**. Otherwise there will be color-coded markers that indicate the status. Snapshot files will have two additional string columns, **StatusName** and **StatusValue**, to indicate the status. In addition to the three values above, **StatusValue** may also be **NotConnected** if ADT was unable to find either the process variable for the data or for the status.

Because of the asynchronous nature of EPICS, there is no guarantee the status displayed in ADT is exactly coordinated with the data, but it should be close. Decreasing the update time may make the status markers more accurately reflect what is happening.

Color Code

Grey40:	Stored data points in the display areas.
Black :	Data points that are not connected (large, square markers)
Grey30:	Data points that are Invalid (large, round markers)
Grey50:	Data points that are OldData (large, round markers)

Xorbit Simulation Mode

Xorbit can in principle be made to simulate the orbit for any of the lattices in the APS, although not all of them have been implemented at this time. To do this there is a special database of process variables in an IOC. These process variables closely resemble the real process variables for monitors and power supplies and have the same names with "**Xorbit:**" attached. When setpoints for magnet power supplies are changed via the control system, Xorbit is notified and calculates the appropriate orbit. It then supplies the new readings to the readbacks for the power supplies and monitors. This process takes about 4 seconds for the over 3000 process variables for the storage ring, but is essentially instantaneous for the PAR.

The simulation includes the effects of random magnet displacements and monitor offsets and includes nonlinear effects from the sextupoles. This simulation capability allows physics application codes to be tested and debugged on something resembling a real system without disturbing the real system or before the real system is operational.

ADT may be run in Xorbit simulation mode by specifying -x on the command line. In this mode it attaches "**Xorbit:**" to the front of all process variable names when it reads them and removes this pattern when it writes them to a file. Consequently, PV files may be tested with Xorbit with no change to the files, provided the lattice has been implemented in Xorbit and provided the actual names and the names in the Xorbit database follow the same naming conventions.

Xorbit calculates magnet strengths in theoretical units (e.g. radians for corrector magnets) whereas the real process variables correspond to power supply currents (Amperes). The relationship between the current and the strength depends on the magnet hardware and the energy of the beam. The Xorbit database has been designed to convert between theoretical units and Amperes, taking these relationships into account.