Core Flight Software Performance Monitor User's Guide

Engineering Directorate
Software, Robotics, and Simulation Division

Baseline
September 2014
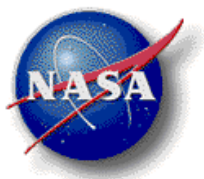
National Aeronautics and Space Administration
Lyndon B. Johnson Space Center
Houston, Texas 77058-3696

Contents

Figures

Tables

1.0   Description

The Core Flight Software (CFS) Performance Monitor, or CPM, is a software tool for analyzing the performance logs generated by the CFS software. A performance log contains information for the performance identifiers (ID) in CFS, including time-stamped events for when the ID becomes active and inactive. CPM displays the contents of a performance log in various formats, including text and plots. Multiple log files can be read; the data is concatenated and treated as a single log. Statistics for each ID, such as minimum and maximum active periods and average event frequency, are calculated on the logged data. CPM is written in Java™, so it can be used on any operating system that supports the Java Runtime Environment (JRE).

Questions or comments concerning this document should be addressed to:

> Software, Robotics, and Simulation Division
> Spacecraft Software Engineering Branch
> Mail Code ER6
> Johnson Space Center
> Houston, TX 77058

2.0   Requirements

CPM is written based on the following Core Flight Executive (cFE), Java, and Java library versions:

- cFE 6.3.2
- JavaSE 1.6
  - Developed in Linux using JavaSE 1.6 and tested on Microsoft Windows and Apple OS X using JavaSE 1.7
- JFreeChart 1.0.18
- JCommon 1.0.22

Compatibility with other versions is not guaranteed. All of the plots generated in CPM are created using JFreeChart. JFreeChart and the associated JCommon library are an open source plotting package for Java (see www.jfree.org/jfreechart for more information). Code used to render the lines in the step and thread of execution plots has been customized to provide capabilities not found in the generic JFreeChart library. The plot library, with customizations, is included in the CPM package.

3.0   Installation

To install CPM copy the Java archive (jar) file CPM.jar to a folder. All Java classes and image files required by the application are contained within this file. The application requires read/write access to the folder it resides in so that log notes file(s) can be created (see paragraph 4.4.3 for further information on log annotations).

To install Java, go to www.java.com and locate the installation instructions appropriate for the operating system on which the application is to be run.

4.0   Operation

To run the application open a command prompt window and navigate to the folder in which the CPM.jar file is installed. At the command prompt type:

```
java –jar CPM.jar
```

Alternately, open a folder browser and navigate to the folder containing CPM.jar, then double-click the CPM.jar icon using the mouse pointer. Note that .jar files must be associated with Java in order for this method to work. The application requires read/write access to the folder from which it is executed in order to create and manage the annotation files for the log data (described in paragraph 4.4.3).

The application can be started with one or more command line arguments using the following format at the command prompt:

```
java –jar CPM.jar [[<- or />]command parameter […]]
```

Each argument consists of a command, optionally preceded by either a '-' or '/', followed by a space, then the command value. The available commands and acceptable values are described in Table 1. The commands can be entered in any order. If a command is entered more than once the last instance's parameter is used. The commands are not case-sensitive, so "-log" is the same as "-LOG", "-Log", etc. The parameter text is not case sensitive unless otherwise noted.

Command	Description	Default
backColor	*Step and thread of execution plot background color	black
exit	`true` to draw the lines between exit and enter events in the step plot; `false` to not draw these lines	true
gaps	`true` to show the gaps between log files in the step and thread of execution plots; `false` to contract the gaps	true
height	Step plot subplot height in pixels	40
hGrid	`true` to show the step and thread of execution plot horizontal (y-axis) grid lines; `false` to hide them	true
id	Performance ID file name. The file name may be case sensitive depending on the operating system	none
idle	`true` to show the CPU idle time in the statistics pie charts; `false` to not include the CPU idle time	true
labels	`true` to show the step plot y-axis tick labels; `false` to hide them	true
laf	"Look and feel" name (e.g., "Nimbus", "Windows", etc.). The names are case sensitive	Metal
log	Performance log file name(s). If multiple names are specified they must be separated by semi-colons (;). Spaces are not allowed. File names may be case sensitive depending on the operating system	none
mainSize	Main application window size in pixels. The parameter format must be in the form *widthxheight* where *width* and *height* are decimal values. A width or height less than the minimum allowed (760 for width, 400 for height) is replaced by the minimum value	760x400
majFrameColor	*Major frame line color	white
majGridColor	*Step and thread of execution plot major grid line color	lightgray
maxFrame	`true` to show the major frame lines in the step and thread of execution plots; `false` to hide them	false
maxID	**Major frame performance ID	none
minFrame	`true` to show the minor frame lines in the step and thread of execution plots; `false` to hide them	false

Command	Description	Default
minFrameColor	*Minor frame line color	gray
minGridColor	*Step and thread of execution plot major grid line color	gray
minID	**Minor frame performance ID	none
order	`name` to sort the performance IDs in the statistics pie charts by ID name; `value` to sort the IDs by statistic value	name
overruns	`true` to show the potential frame overrun indicators in the step and thread of execution plots; `false` to hide them	true
plot	The initially selected plot tab: Step, Thread, or Statistics. If a log is loaded this determines which plot is initially displayed	step
shading	`true` to shade the area beneath the plot lines in the step plot; `false` to remove the shading	true
spacing	Step plot subplot spacing in pixels	5
type	`pie` to display the statistics data if pie chart format; `bar` to display the data in bar chart format	pie
vGrid	`true` to show the step and thread of execution plot vertical (x-axis) grid lines; `false` to hide them	true

*   Colors must be one of those recognized by Java (black, blue, cyan, darkgray, gray, green, lightgray, magenta, orange, pink, red, white, or yellow), or in hexadecimal format as *0xRRGGBB* where R, G, and B are hexadecimal digits representing the red, green, and blue color components (all 6 digits are required). The '0x' is optional

** The ID must be in hexadecimal format. A preceding '0x' is optional. The ID may be no more than 8 digits and the high bit may not be set

Table 1. Command line arguments

For example, the following command loads two log files, a performance ID file, and sets the thread of execution plot as the initial plot to display with a white plot background and the data gap between the log files suppressed:

```
java -jar CPM.jar –log logFile;../logs/anotherLogFile –id
   /user/abc/idFile –plot thread –background white –gaps false
```

An invalid command or command parameter results in program termination. An invalid parameter displays an error message at the command prompt. An invalid command or a valid command without an associated parameter produces the following output at the command prompt:

```
usage:
java -jar CPM.jar [[<- or />]cmd parameter [...]]
 Command line arguments:
  Description            Command      Value
  --------------------- ------------ ------------------------------------
  Plot background color backColor    <[name] or [0xRRGGBB]>
  Show step exit lines  exit         <true or false>
  Show data gaps        gaps         <true or false>
  Step plot height      height       <step plot height>
  Show horizontal grid  hGrid        <true or false>
  Load ID file          id           <ID file name>
  Show CPU idle time    idle         <true or false>
```

6

```
Show y-axis tick labels labels        <true or false>
Load look & feel         laf          <look & feel>
Load log file(s)         log          <log file name[;log file name 2[...]]>
Main window size         mainSize     <widthxheight>
Show major frame lines   majFrame     <true or false>
Major frame line color   majFrameColor <[name] or [0xRRGGBB]>
Major grid line color    majGridColor <[name] or [0xRRGGBB]>
Major frame ID           majID        <0x########>
Show minor frame lines   minFrame     <true or false>
Minor frame line color   minFrameColor <[name] or [0xRRGGBB]>
Minor grid line color    minGridColor <[name] or [0xRRGGBB]>
Minor frame ID           minID        <0x########>
Stats plot sort order    order        <name or value>
Show frame overruns      overruns     <true or false>
Selected plot tab        plot         <step or thread or statistics>
Shade beneath steps      shading      <true or false>
Step plot spacing        spacing      <step plot spacing>
Stats chart type         type         <pie or bar>
Show vertical grid       vGrid        <true or false>
```

A detailed explanation of the files, L&F, and plot preferences alterable by command line argument can be found in subsequent paragraphs of this document.

Once the application is executed the CPM main window appears as shown in Figure 1. The graphical user interface (GUI) "look and feel" (L&F) can be selected by the user from a list of ones installed on the operating system. If the L&F is changed then the application window and dialogs may differ in appearance (but not function) from those shown in the figures below. See paragraph 4.3.4 on how to alter the L&F.
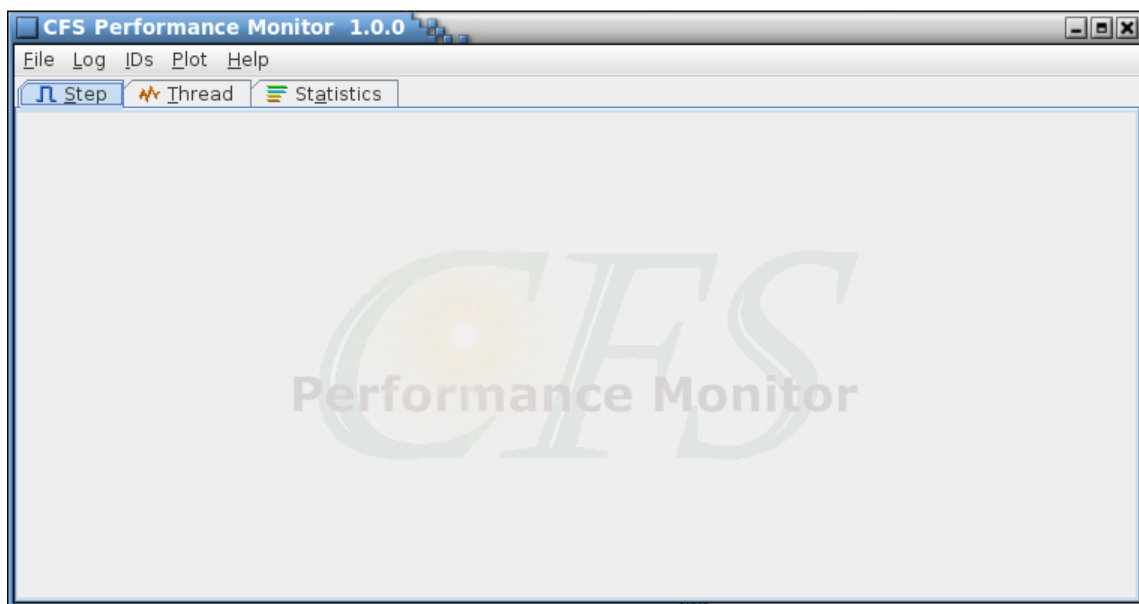

Figure 1.   CPM main window

The main window header contains the program name and version number. The main window is divided into a menu bar along the top and a tabbed plot display area underneath. The window can be resized as desired. Each menu contains one or more menu items or sub-menus. A menu items that is grayed-out indicates that the affected item is not available at that time. This item become active once a log file and/or ID file is loaded. A description of each of the menu items is provided in subsequent paragraphs.

The three tabs are for choosing the plot for display. Selecting a tab is equivalent to using the associated **Plot** menu item; see paragraph 4.6 and sub-paragraphs for further details.

4.1   Program Preferences

The programs preferences are stored in a location dependent on the operating system and are updated as needed by the CPM application. For example, the Windows operating system stores the preferences in the system registry under the key name:

```
HKEY_CURRENT_USER\Software\JavaSoft\Prefs\/C/F/S/Performance/Monitor
```

In Linux the preferences are stored in the file:

```
/<user home directory>/.java/.userPrefs/CFSPerformanceMonitor/prefs.xml
```

The user should have no need to manually edit these preferences; however, a description of the preference keys and associated values is provided below for reference purposes.

CONFIG< *name*>   *name* is a unique name, chosen by the user, to identify the configuration (the file can contain more than one configuration). The value consists of the names of the configuration's ID and log files, including the files' full path names. If multiple log files are listed then the names are separated by a semicolon (;)

LookAndFeel   The name of the selected "look and feel". See paragraph 4.3.4 for more information on the application look and feel

PerfIDListFile   The name of the latest active performance ID file, including the file's full path name

PerfLogFile   The name(s) of the latest active performance log file(s), including each file's full path name. If multiple log files are listed then the names are separated by a semicolon (;)

SearchFileExtension   The file extension of the files to be searched for performance IDs. The extension is not case sensitive. The default file extension is `.h`

SearchPerfIDSuffix   The character string that defines a performance ID in the source code. An ID is assumed to end with this character string. The string is case sensitive. The default suffix is `PERF_ID`

SearchSourcePath   The path to the targeted source code. This is used when automatically generating the list of performance IDs and ID names from those found in the source code

SearchSubfolders   The subfolders within the source code path that are to be searched for files containing performance IDs. The subfolder names are not case sensitive. The default search subfolders are `apps` and `cfe`

8

PlotPreferences   The plot preferences values, separated by commas (,). The preferences, valid ranges, and default values are shown in Table 2 (the order in the table is the same as the order that the preferences are stored). A detailed description of the preferences can be found in paragraph **Error! Reference source not found.**

Value	Range / Units	Default
Step plot height	20 - 100 pixels	40
Step plot spacing	0 - 15 pixels	5
Show step plot labels	true or false	true
Show step plot shading	true or false	true
Show plot horizontal grid lines	true or false	true
Show Plot vertical grid lines	true or false	true
Show plot minor frame lines	true or false	false
Show plot major frame lines	true or false	false
Minor frame ID	performance ID in hexadecimal	-1
Major frame ID	performance ID in hexadecimal	-1
Include CPU idle time	true or false	true
Statistics sort order	0 = sort by name; 1 = sort by value	0
Show plot data gaps	true or false	true
Plot background color	hexadecimal RGB color value (0xRRGGBB)	0x000000
Major grid line color	hexadecimal RGB color value (0xRRGGBB)	0xc0c0c0
Minor grid line color	hexadecimal RGB color value (0xRRGGBB)	0x808080
Minor frame line color	hexadecimal RGB color value (0xRRGGBB)	0x808080
Major frame line color	hexadecimal RGB color value (0xRRGGBB)	0xffffff
Show frame overruns	true or false	true
Statistics chart type	0 = pie chart; 1 = bar chart	0
Show step plot exit lines	true or false	true

Table 2. Plot preferences

4.2 Mouse and Keyboard Navigation

The application's menus, dialogs, and GUI components can be manipulated using the mouse pointer, mouse buttons, and mouse wheel, as well as with the keyboard. Keyboard mnemonics are provided for the menu items, tab headings, and dialog buttons. These are accessed by pressing the Alt key in addition to another key; i.e., Alt+*key*, where *key* is the underlined character in the menu, tab, or button text (the key case is ignored). For example, pressing Alt+F or Alt+f in the main application window opens the **File** menu. The Tab and arrow keys can be used to navigate between the components in a dialog or window, and the pressing the Enter or space key actuates a control (e.g., a button or check box).

When a dialog containing a table is initially displayed it has no row selected. A row can be selected by positioning the mouse pointer over a cell in the row and pressing the left mouse button, or by using the keyboard. To select an initial row with the keyboard press the Enter or Space key when the table has the keyboard focus (which it does initially); this selects the table's topmost visible row and sets the focus to that row's leftmost cell. The up and down arrow keys can then be used to change the selected row and the left and right arrows can change the cell focus. The selected row is highlighted, with a darker highlight appearing in the cell with the focus. Multiple, consecutive rows can be selected using a

combination of the mouse/keyboard and the Shift key. Highlight the starting row, then either (a) continue to press the left mouse button and drag the pointer, (b) hold the Shift key and left-click the mouse on another row (the two rows, plus any in between, are highlighted), or (c) hold the Shift key and press the arrow key to highlight as many rows as desired. Individual rows can be selected/deselected by pressing the Ctrl key and selecting the row with the mouse (selected rows must be consecutive). The entire table may be selected by pressing Ctrl-A. Once one or more rows are selected the highlighted data can be copied by pressing Ctrl-C. To paste the data into another application (e.g., spreadsheet or text document) use the Ctrl-V key sequence.

Details specific to navigation in certain windows and dialogs are provided in the components' descriptions in later sections.

4.3   File Menu

The File menu provides selections for manipulating application configurations, updating the application's overall appearance, and exiting the program. A configuration consists of an ID file name and one or more log file names. The information for each saved configuration is stored in the program preferences location.

4.3.1   Load Config

If no configurations have been saved then this item is grayed out and cannot be selected. If one or more configurations have been saved then when this item is selected the Load Config dialog is displayed (see Figure 2). The drop-down field contains a list of stored configurations listed in alphabetical order. The mouse or keyboard can be used to navigate through the dialog. The ID and log files referenced by the currently displayed configuration are shown below the drop-down field.
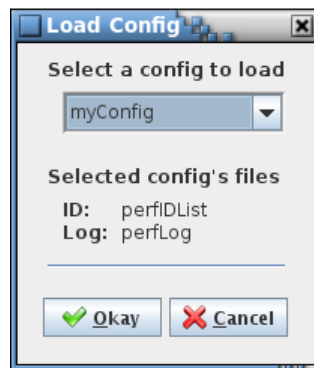
Figure 2.   Load Config dialog

Select the **Okay** button to load the selected configuration's files. The mouse pointer changes while processing the configuration change. Once loaded, the data is plotted immediately. The currently selected plot tab determines which plot is displayed. If data is already plotted then the plot is updated with the new log and ID data. Select **Cancel** to exit the dialog without loading a configuration.

See paragraphs 4.4.1 and 4.5.1 for information regarding log and ID files.

4.3.2   Save Config

This menu item remains grayed out (non-selectable) until an ID and log file are loaded. If an ID and log file have been loaded then the Save Config dialog is displayed (see Figure 3). The editable drop down field initially shows the name of the latest configuration loaded since the application started; if no configuration has been loaded the drop-down field is blank. The ID and log files currently loaded are displayed below the drop-down field. Select a name in the drop-down field to overwrite an existing configuration, or type a new, unique name to create a new configuration. Select the **Okay** button to save the configuration's ID and log file names. Select **Cancel** to exit the dialog without saving the configuration.

Figure 3.   Save Config dialog

4.3.3   Delete Config

This menu item can only be selected if at least one configuration is stored in the program preferences location; otherwise the item is grayed out. The dialog displayed (see Figure 4) contains a drop-down field with an alphabetized list of the stored configurations. When a configuration is chosen the ID and log file names are displayed beneath the drop-down field. Select **Okay** to delete the chosen configuration stored in the program preferences location. Note that this does not alter the currently loaded log and ID files. Select **Cancel** to exit the dialog without deleting a configuration.
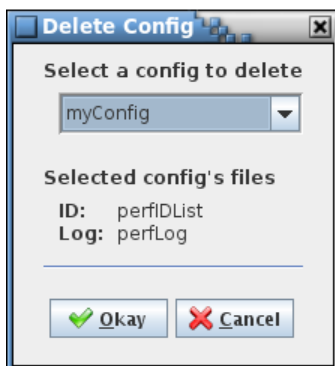

Figure 4.   Delete Config dialog

4.3.4   Appearance

The Application Appearance dialog (Figure 5) allows choosing the style, or "look and feel" (L&F), applied to the program's GUI components. Different L&Fs can change the shape and color scheme of the graphical components (see Figure 6). The default is "Metal", the standard L&F provided with Java. The actual list of L&F selections displayed in the dialog is dependent on the available L&Fs loaded on the host machine. When the radio button associated with the desired L&F is selected the Application Appearance dialog, main application window, and any other open CPM dialogs immediately are redrawn to reflect the L&F chosen. Select the **Close** button to exit the dialog.
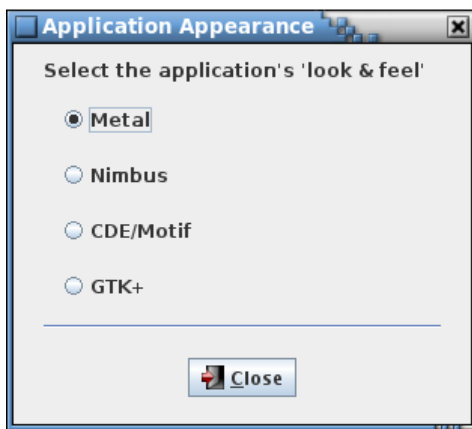
Figure 5.   Application Preferences dialog
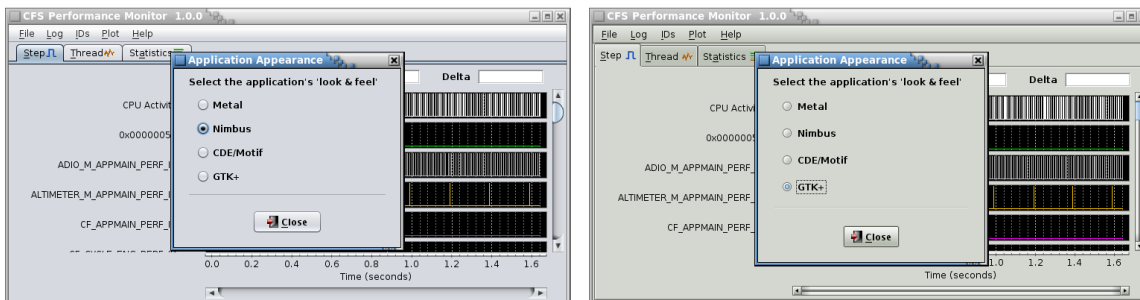
Figure 6.   Example look and feel differences

4.3.5   Exit

A dialog appears so that the user can confirm whether or not to exit the application. Select **Okay** to exit CPM – the main windows and any open dialogs are closed. Select **Cancel** to close the dialog without exiting the application.

4.4   Log Menu

The Log menu contains selections for loading and reviewing log data.

4.4.1   Read Log

If log data is currently loaded, via this command or the **Load Config** command, then a dialog appears asking if the current data should be appended to, or discarded in favor of the log(s) to be loaded. If the

Append button is selected then the new log data is added to that already loaded. Any newly selected log matching one already loaded is ignored (i.e., the data is only loaded once for each log selected). If the **Clear** button is selected the current data is cleared before loading the new log(s). Select **Cancel** to leave the loaded log data unchanged.

If **Append** or **Clear** is chosen, or if no log data has yet been loaded, a file choice dialog appears allowing the user to select one or more performance log files to process. Log file names may contain spaces if the name is placed within double quotes in the dialog's file name text field. Multiple log files can be loaded, either by repeated use of the **Read Log** command (and selecting **Append**), or in groups by using the mouse pointer to select the files from the list shown in the file choice dialog or by typing in the file name in the text field provided. The file names must be separated by spaces or commas. When multiple log files are loaded they are concatenated based on the log header time stamps. Once a log file is loaded the remaining grayed out menu items in the **Log** menu become active, as well as those in the **Plot** menu.

Once a log is loaded a dialog appears showing the number of unique performance IDs referenced in the log (see Figure 12), then a consistency check is done on the data. For a given performance ID, entry and exit events should occur in alternating pairs. The consistency check looks for and notes the total instances of consecutive entry or exit events for the same performance ID. Also, events should be logged in the order of their time stamps, so the number of instances of an event occurring out of time sequence is noted.

If any inconsistencies are found a dialog appears (see Figure 7) indicating the number and type of inconsistencies detected. If multiple logs are loaded then the total of the inconsistencies is shown (note that when loading multiple logs, instances of consecutive entry/exit events can be expected since a time gap between the logs, and hence missed events, occurs). Consecutive events are indicated in the log data dialog; see paragraph 4.4.2. Select the **Okay** button to close the Log Inconsistency dialog.
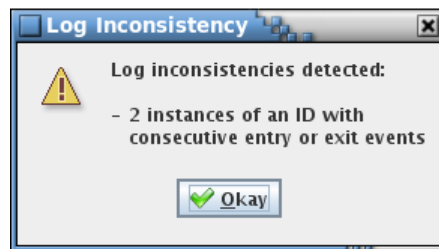
Figure 7.   Example log inconsistency dialog

Neither consecutive events nor time inconsistencies affect plotting of the logged events. An event is plotted in order of its time stamp regardless of where it appears in the log file.

4.4.2   Show Header

A dialog appears providing the contents of the log file header (see Figure 8). The header contains information on the conditions when the log file was created, such as the time stamp, endianess, and the number of log entries ("Data count"). If multiple log files are loaded then each has a separate tab in the log header dialog, with the tab showing the log file name associated with the tab's header information. If the mouse pointer is paused over the tab a tool tip window appears showing the full path and file name of the log file. The two time parameters shown in the dialog, Time (seconds) and Time

(subseconds), are used to determine the relative positioning of the log data when multiple log files are loaded.

Figure 8.   Show Header dialog

The filter and trigger mask values are arranged in the order of highest to lowest bit. For example, the first bit (representing performance ID 0x00000000) is displayed in the last mask value's lowest bit and the bit for ID 0x00000021 is displayed in the next to the lowest bit of the next to the last mask value (i.e., the thirty fourth bit); see below.

```
…        last mask value – 1                    last mask value

…   f   f   f   f   f   f   f   d   f   f   f   f   f   f   f   e
… 1111 1111 1111 1111 1111 1111 1111 1101  1111 1111 1111 1111 1111 1111 1111 1110
…                                    ^                                          ^
                              Bit for 0x00000021                     Bit for 0x00000000
```

The log header table can be output to a printer or file by pressing the **Print** button. A dialog appears allowing selection of the target printer or the option to output the table to a file. Other options may include the number of copies, page orientation, page margins, etc. depending on the host operating system. If multiple log files are loaded only the currently displayed tab is printed. Select the **Close** button to exit this dialog.

4.4.3   Show Data

Once one or more log files are loaded the **Show Data** menu item becomes available. Selecting this menu item displays the log data dialog (see Figure 9). This dialog may remain open while other actions are performed via the main window. Selecting the **Print** button sends the contents of this table to the printer or file selected in a subsequent dialog (warning: the number of pages generated can be very large). Selecting the **Close** button removes the dialog from the display. The dialog can be resized as desired, though all columns except the **Name** and **Notes** columns are of fixed widths.

Figure 9.   Show Data dialog

The log data dialog window frame indicates the name of the log file. If multiple logs are loaded then each file name is displayed, separated by commas. The table is divided into eight columns, described in Table 3. Descriptive tool tips appear by placing the mouse pointer over the table column headers and pausing briefly.

Column	Description
Index	Sequential index of the event in the log
ID	Event performance ID in hexadecimal
Name	Event performance ID name. If no name has been assigned to the ID then "*** Undefined ***" is displayed

Column	Description
Time Stamp (sec)	Event time stamp in seconds. The times are relative to the first event in the earliest log file, so the first event always shows a time stamp of 0.000000 seconds. If multiple logs are loaded then the log's time stamp (see the log header description in paragraph 4.4.2) determines the time gap between the events in different files. However, if the data gaps are hidden (see paragraph 1.0 concerning the "Show data gaps" preference) then the time difference between events from different files is removed; the gap between the last event in a log and the first event in the next log is reduced to a single clock tick. The log header contains the value for the number of clock ticks per second, which inverted is the time for a single tick
Entry/Exit	Displays the icon of a running figure in green ( ) for an entry event and a standing figure in red ( ) for an exit event. The first bit in the performance ID (shown in the **ID** column) determines whether or not an event is an entry event (first bit = 0) or an exit event (first bit = 1)
Seq Err	If a yellow triangular icon ( ) appears in a row of this column it indicates that consecutive events have occurred of the type shown in the adjacent **Entry/Exit** column for the ID shown in same row's **ID** column
Overrun (sec)	Displays the amount of time, in seconds, that this entry or exit event exceeds the user-defined allotted time since this ID's previous entry event. The inverse of the ID frequency determines the time limit. Example: If the ID frequency is set to 100 Hz and the time between consecutive entry events is 0.012 seconds then the **Overrun** column displays 0.002 (= 0.012 − (1 / 100))
Notes	User-supplied annotation for this log entry. The notes can be edited. The text automatically word-wraps based on the column width, which is adjustable

Table 3. Log content dialog columns

The data can be sorted in ascending or descending order of the data in a column by selecting the column header with the mouse pointer – an arrow appears in the header indicating the sort direction. Select the **Index** or **Time Stamp** columns to restore the data to time sequential order.

A log event can be annotated if desired. These notes are appended to the tool tips that appear when the mouse pointer hovers over a point in the step or thread of execution plot (see Figure 20). Annotations are saved in a text file in the directory in which the application is started. Each log file has its own annotation file which is created when the first annotation is added to the table. The file name is derived from the log file's internal time tag so that the name is unique for each log file. When a log is loaded its annotation file, if present, is read and the annotations are added to the log data. The annotation file is updated automatically when the text in the Notes column is altered. If all of the Notes column entries are erased the annotation file for the affected log is deleted as well. To add or alter an annotation double left-click on the target cell in the **Notes** column (alternately, maneuver the keyboard focus to the target cell and press the Enter or Space key). A text editor dialog appears (see Figure 10) containing a copy of the text in the cell, if any. Use the keyboard to alter the text. The editor can be resized and allows entering line feed characters into the annotation. To enter the updated text into the table press the **Okay** button, or press the **Cancel** button to ignore any changes.
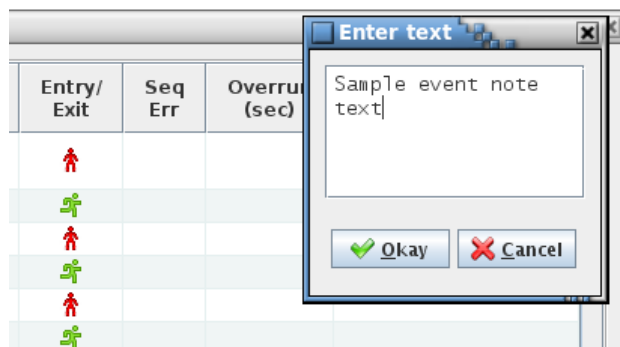
Figure 10. Event annotation

The log data table can be output to a printer or file by pressing the **Print** button. A dialog appears allowing selection of the target printer or the option to output the table to a file. Other options may include the number of copies, page orientation, page margins, etc. depending on the host operating system.

4.4.4   Show Statistics

Once one or more log files is loaded the **Show Statistics** menu item becomes available. Selecting this menu item displays the log statistics dialog (see Figure 11). The statistics dialog may remain open while other actions are performed via the main window. Selecting the **Close** button removes the dialog from the display. The dialog can be resized as desired, though all columns except the **Name** column are of fixed widths.

Note: If multiple log files are loaded then the data gaps between the log files are ignored *when computing the statistics values.*

The log statistics dialog window frame indicates the name of the log file. If multiple logs are loaded then each file name is displayed, separated by commas. The dialog contains two tables, the upper one for individual performance ID statistics, and a lower one for combined statistics. Descriptive tool tips appear by placing the mouse pointer over the table column headers and pausing briefly.

Figure 11. Show Statistics dialog

The data can be sorted in ascending or descending order of the data in a column by selecting the column header with the mouse pointer – an arrow appears in the header indicating the sort direction. The table column definitions are described below in Table 4.

17

Column	Description
ID	Event performance ID in hexadecimal
Name	Event performance ID name. If no name has been assigned to the ID then "*** Undefined ***" is displayed
Entry Events	Total number of entry events logged for this ID. For the **Overall** row this is the total number of entry events for all IDs
Exit Events	Total number of exit events logged for this ID. For the **Overall** row this is the total number of exit events for all IDs
Avg Freq (evt/sec)	Average frequency, in events/second, at which entry events occur. The **Overall** row shows the average frequency of all events for the duration of the log data
Time On (sec)	Total time, in seconds, that this ID is active; i.e., between consecutive entry and exit events. For the **Overall** row this is total number of seconds when at least one ID is active
Time Off (sec)	Total time, in seconds, that this ID is inactive; i.e., between consecutive exit and entry events. For the **Overall** row this is total number of seconds when no ID is active
Time On (%)	Total percentage of the time that this ID is active; i.e., between consecutive entry and exit events. For the **Overall** row this is total percentage of time when at least one ID is active
Time Off (%)	Total percentage of the time that this ID is inactive; i.e., between consecutive exit and entry events. For the **Overall** row this is total percentage of time when no ID is active
Min On (sec)	Minimum time, in seconds, that this ID is active; i.e., between consecutive entry and exit events. "n/a" is displayed if no events occur for this ID. The **Overall** row shows the smallest of the individual minimum times ("n/a" times are ignored)
Max On (sec)	Maximum time, in seconds, that this ID is active; i.e., between consecutive entry and exit events. "n/a" is displayed if no events occur for this ID. The **Overall** row shows the largest of the individual maximum times ("n/a" times are ignored)
Min Int (sec)	Minimum time, in seconds, between consecutive entry events. "n/a" is displayed if no events occur for this ID. The **Overall** row shows the smallest of the individual minimum intervals ("n/a" times are ignored)
Max Int (sec)	Maximum time, in seconds, between consecutive entry events. "n/a" is displayed if no events occur for this ID. The **Overall** row shows the largest of the individual maximum intervals ("n/a" times are ignored)
Min Over (sec)	Minimum frame overrun, in seconds. "n/a" is displayed if no overruns occur for this ID. The **Overall** row shows the smallest of the individual frame overruns ("n/a" times are ignored)
Max Over (sec)	Maximum frame overrun, in seconds. "n/a" is displayed if no overruns occur for this ID. The **Overall** row shows the largest of the individual frame overruns ("n/a" times are ignored)

Table 4. Statistics dialog columns

Placing and pausing the mouse pointer over one of the minimum or maximum statistics cells (Min On, Min Off, Min Int, Max Int, Min Over, or Max Over) causes a tool tip to appear that shows the time at which the minimum or maximum value occurred. For the Overall table the ID responsible for the minimum/maximum is also shown.

The statistics data table can be output to a printer or file by pressing the **Print** button. A dialog appears allowing selection of the target printer or the option to output the table to a file. Other options may include the number of copies, page orientation, page margins, etc. depending on the host operating system.

4.4.5   Show ID Count

The Show ID Count command displays a dialog (see Figure 12) showing the number of unique performance IDs referenced in the log data. This dialog is automatically displayed when a log is read.
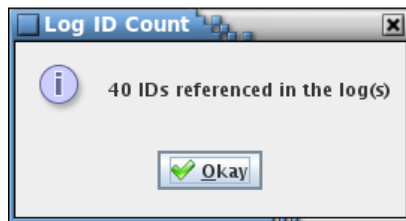
Figure 12.   Log ID Count dialog

4.5   IDs Menu

4.5.1   Load from File

Selecting this command results in a dialog being displayed asking the user for the path and file name containing a list of performance IDs and ID names. This file is created via the **Save to File** command described in paragraph 4.5.2. If a list of IDs is already in use, e.g., by reading a log or after a previous use of this command to load a list of IDs, then a dialog first appears asking if the current list should be appended to, or discarded in favor of the one to be loaded. If the **Append** button is selected then only new IDs are added to the list, and if **Clear** is selected the current list is cleared. Select **Cancel** to leave the ID list unchanged.

Once a file is selected and the **Load** button pressed a dialog appears showing the number of IDs loaded. If IDs are appended to and existing list then the dialog also indicates the number of new IDs added. If the log content or statistics dialog is open the ID name column is automatically updated with any ID or name changes. If the step or thread of execution plot, or statistics bar chart is currently displayed then the y-axis labels are updated to show the ID names. If a statistics pie chart is visible then the pie segment labels are updated. Selecting the **Cancel** button leaves the ID list unchanged.

4.5.2   Save to File

When this command is selected a dialog appears asking the user to provide a path and file name in which to save the current list of performance IDs and ID names. The file is in text format with each ID's information occupying a single line in the file. The format of an ID entry is as follows:

```
ID name,ID,color,frequency,annotation
```

The ID and color values are in hexadecimal.

Once saved, the IDs can be then be loaded from this file whenever needed using the **Load from File** command (see paragraph 4.5.1).

4.5.3 Edit List

The ID editor dialog (see Figure 13), accessed by selecting the **Edit List** command, allows for adding or removing performance IDs, altering the ID names, setting the ID's plotted color and expected frequency, and providing explanatory notes. It also lets the user choose which IDs to display and in what order in the plots.
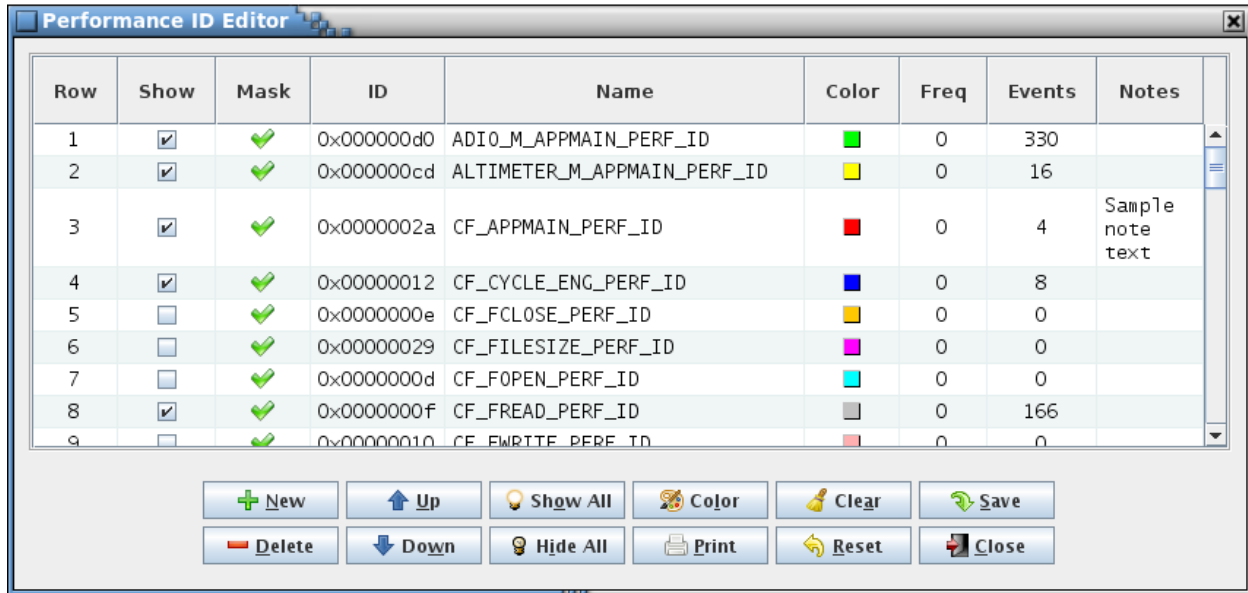
Figure 13. Performance ID Editor dialog

If IDs exist (for example, by loading a log or a saved ID list) then the table is automatically populated. If no IDs are loaded then the table is initially empty.

Left-clicking a column header with the mouse causes the table to be sorted by the contents of the column. Click once to sort in ascending order, and click again to sort in descending order. A small arrow icon appears beside the column name indicating the sort direction. The **Row** column cannot be sorted on; its values are unaffected by sorting of the other columns.

The columns in the dialog's table are described in Table 5.

Column	Description
Row	Table row number. The row numbers are fixed even if the table rows are rearranged, or if rows are added or deleted. If a warning dialog appears during editing (due to an invalid input) then the row number is shown to aid in locating the error
Show	This column displays a check box that determines if an ID is plotted (checked) or not (unchecked). The mouse or keyboard (Enter or Space key) can be used to toggle the state
Mask	Indicates if the ID is enabled in the filter mask of at least one of the currently loaded logs by displaying a green check (✔) or disabled by displaying a red X (✖). An ID that is disabled in the filter mask cannot have any of its events logged. The filter mask must be set in CFS prior to recording the log data; if no mask is explicitly set then all events are included in the mask

Column	Description
ID	Performance ID in hexadecimal. The ID can be edited
Name	Event performance ID name. If no name has been assigned to the ID then the cell is blank. The name can be edited
Color	Displays a colored rectangle that represents the color used when plotting the ID in the step, thread of execution, and statistics plots. Selecting a rectangle via the mouse or keyboard causes a color selection dialog to appear, allowing the ID color to be changed
Freq	ID frequency in events/second. A frame overrun is deemed to have occurred if the time between the ID's consecutive entry events or between an entry-exit event pair exceed the inverse of the frequency. Overruns are displayed in the Show Data dialog (see paragraph 4.4.2), and in the step (see Figure 21) and thread of execution (see Figure 23) plots. A frequency value of zero indicates no frame overrun check is made for this ID
Events	Total number of Entry and Exit events logged for this ID. Zeroes are displayed for all IDs if no log file is loaded
Notes	User-supplied annotation for this performance ID. The notes can be edited. The text automatically word-wraps based on the column width, which is adjustable

Table 5. ID editor dialog columns

The table cell displayed in a slightly darker highlight color has the keyboard focus. Left-click a cell or use the arrow keys to change which cell has the focus.

When a log file is loaded the **Show** flag is automatically enabled for all IDs referenced within the log; any unreferenced IDs defined in the ID list have the **Show** flag cleared. Left-clicking the check box with the mouse toggles the flag state. The keyboard can also be used by maneuvering the keyboard focus to the desired check box and pressing the Enter or Space key to toggle the selection. The **Show All** and **Hide All** buttons allow for enabling and disabling all of the IDs' **Show** flags. If an ID is enabled but has no data in the log file then the ID is shown in the plots but with no data displayed. If all of the IDs are disabled then the plots show only the CPU idle time (which is 100% with no IDs selected).

The **ID**, **Name**, **Freq**, and **Notes** cell values can be edited. Double left-click on the target cell to enable editing (alternately, maneuver the keyboard focus to the target cell and press the Enter or Space key).

For the **ID**, **Name**, and **Freq** cells use the keyboard to alter the cell contents. To enter in the updated value press the Enter key or move the focus to another cell using the mouse - until this is done the value hasn't been updated in the table. The **ID** cell only allows integer or hexadecimal values (hexadecimal values must be preceded by '0x'); if a non-integer and non-hexadecimal value is entered then an error dialog appears and the original cell value is restored. A valid value is displayed in hexadecimal. The **ID** and **Name** cell values must be unique for each ID. If the ID (name) matches the ID (name) in another row then an error dialog appears and the cell contents reverts to its original value. An exception is the case when the **Name** cell is blank since IDs are allowed to have no name. Under this circumstance the ID itself, and not the name, is used when labeling the plots. The **Freq** value must be a non-negative number (fractional values are allowed); anything else causes an error dialog to be displayed and the original value to be restored.

An ID can be annotated if desired. Annotations are stored in the ID list when saved via the **Save to File** command (see paragraph 4.5.2). To add or alter an annotation double left-click on the target cell in the **Notes** column (alternately, maneuver the keyboard focus to the target cell and press the Enter or Space key). A text editor dialog appears (see Figure 14) containing a copy of the text in the cell, if any. Use the keyboard to alter the text. The editor can be resized and allows entering line feed characters into the annotation. To enter the updated text into the table press the **Okay** button, or press the **Cancel** button to ignore any changes.
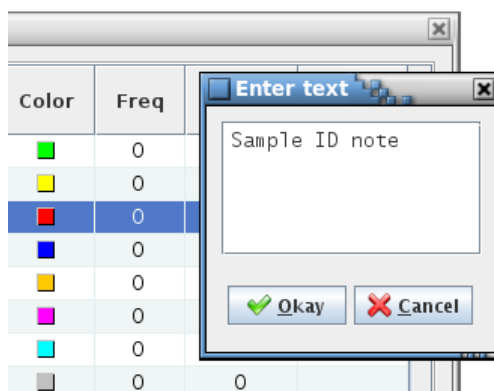
Figure 14. ID annotation

The IDs, when plotted in the step and thread of execution plots, and the statistics charts, can be distinguished by color. ID colors can be adjusted by selecting an ID's color box in the **Color** column, using the mouse (left-click) or keyboard (press Enter or Space key once the color cell has the focus). A color choice dialog is displayed. The color selection method(s) provided by this dialog varies based on the current look and feel – see Figure 15 for an example.

Figure 15. Example color choice dialog

A preview panel (located at the right side of the dialog) shows the original color (left box) and the currently selected color (right box). Press **Okay** to choose the selected color, or **Cancel** to exit the dialog without changing the ID color. The ID's color box in the ID editor changes to reflect the color chosen.

A new ID can be inserted by highlighting a row in the table and selecting the **New** button. A blank row is inserted above the highlighted row (if the table is empty and **New** is selected then the new row appears as row 1 in the table). The **Show**, **ID**, **Name**, **Color**, **Freq**, and **Notes** cells can then be edited as described above.

One or more rows can be removed from the list by highlighting the row(s) and selecting the **Delete** button. The editor prevents deletion of any IDs that are referenced in the loaded log (any ID that shows a non-zero value in the **Events** column). Attempting to do so results in a warning message and if multiple rows were selected for deletion then the IDs that were skipped remain highlighted.

The order of the rows in the table determines the order that the IDs appear in the plots. The table row order can be changed using the column sorting described earlier or by highlighting one or more rows and selecting the **Up** or **Down** buttons. Each press of the button moves the selected row(s) up or down one position relative to the unselected rows.

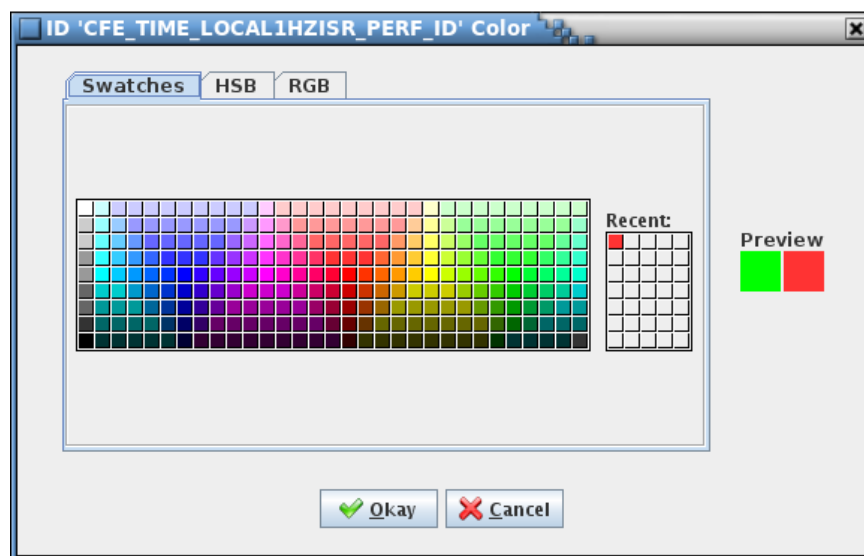ID color is automatically assigned when an ID is added to the list (the color can be changed as described previously). If an ID is inserted using the **New** button the color defaults to white. When a log is read then any ID not already defined in the ID list has a color assigned from a pre-defined list. The colors used are green, yellow, blue, orange, magenta, cyan, light gray, and pink, and are assigned in that order. The editor allows the user to reassign these colors to the IDs by pressing the **Color** button. A confirmation dialog appears, and if **Okay** is selected the IDs are assigned a color in the order previously mentioned based on the current ordering of the IDs in the list. In other words, if the list is rearranged (e.g., by using the move up or down buttons, or by sorting via the column headers) then the color s are assigned based on the new arrangement. The IDs' color boxes in the editor change to reflect the new colors.

The ID table can be output to a printer or file by selecting the **Print** button. A dialog appears allowing selection of the target printer or the option to output the table to a file. Other options may include the number of copies, page orientation, page margins, etc. depending on the host operating system.

The table may be cleared by pressing the **Clear** button. A confirmation dialog appears and if **Okay** is selected all of the rows in the table are deleted. If the cleared list is saved and log data is loaded then the IDs contained in the log are restored to the list once the editor is reopened. Select **Cancel** to return to the editor without changing the table.

Any changes to the ID list must be saved in order to be applied to the plots. Pressing the **Save** button stores the changes. Empty rows are ignored and not saved. Pressing the **Close** button exits the ID Editor dialog. If there are any unsaved changes a dialog appears asking for confirmation to discard the changes. If **Okay** is selected then the unsaved changes are lost; select **Cancel** to return to the editor.

The **Reset** button restores the ID list to its original order and removes any changes, including any added or removed rows, and updates to the IDs and ID names. The effect is the same as closing the editor without saving the changes, then opening it again.

The dialog button functions and shortcut keys are summarized in Table 6.

Button	Shortcut	Description
New	Alt+N Insert	Insert a blank row above the currently selected table row
Delete	Alt+D Delete Ctrl+X	Delete the currently selected table row(s). Rows containing IDs with logged events cannot be deleted
Up	Alt+U Ctrl+↑	Move the currently selected row(s) up one row
Down	Alt+W Ctrl+↓	Move the currently selected row(s) down one row
Show All	Alt+O	Enables the Show state for all of the IDs in the table
Hide All	Alt+I	Disables the Show state for all of the IDs in the table
Color	Alt+L	Auto-assign colors to the IDs based on the current order of the IDs in the table
Print	Alt+P	Output the ID table to a printer
Clear	Alt+A	Clear the contents of the ID table
Reset	Alt+R	Restore the ID list to the initial values and order, removing all changes
Save	Alt+S	Saves and changes made to the ID list
Close	Alt+C	Exit the ID editor dialog

Table 6. ID editor dialog buttons

4.5.4   Clear List

When selected a confirmation dialog box appears. If **Okay** is chosen then the currently loaded list of performance ID is erased. If a log file is currently loaded then the performance ID hexadecimal values are retained; only the information (e.g., names, frequency, notes) and any IDs that have been added via the ID editor are erased. If the log content or statistics dialog is open the ID names in the **Name** column is automatically replaced by "*** Undefined ***". If the step or thread of execution plot, or statistics bar chart is currently displayed then the y-axis labels are replaced with the ID hexadecimal values. If a statistics pie chart is visible then the pie segment labels are replaced with the ID values. If **Cancel** is chosen then the list of IDs is unaffected.

4.5.5   Search Source

This option searches through the source code located in the path name specified by the user for performance ID declarations. From this a list of IDs with associated ID names is built. Normally the source code to be searched would be that contained in the CFS cFE and applications subfolders; however, there is no restriction on the folder location to search.

If an ID list is currently loaded then a dialog appears asking the user to choose to append IDs to the existing list (**Append** button), clear the existing list prior to the search (**Clear** button), or abandon the search (**Cancel** button). If **Append** is selected only new IDs are added; IDs already in the list are ignored.

A dialog then appears allowing the source code path to be chosen. Selecting the **Cancel** button in this dialog exits the search; pressing **Okay** targets the currently selected folder for searching.

Next, a dialog appears that allows the user to adjust the search criteria, including the subfolders within the source path to search, the extension of the files to check, and the character string that identifies a performance ID definition. See Figure 16, which shows the default values for these criteria. If the target

folder contains no subfolders then all of the files within the target folder are automatically searched; the dialog indicates this condition. The extension is not case-sensitive, but the ID suffix is. Pressing the **Default** button restores the criteria to the default values. Selecting **Okay** continues with the search using the criteria specified. Any changes made to the search criteria are saved in the program preferences location. The search in canceled if the **Cancel** button is selected.
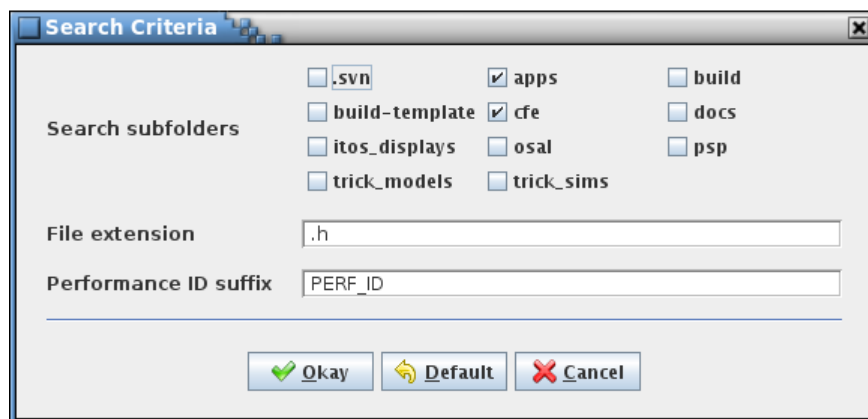
Figure 16. Search Criteria dialog

During the search a progress bar appears showing the folder currently being searched. The progress bar is removed when the search completes. Files with names ending in the indicated extension characters and located within the listed subfolders of the target source code folder are searched for lines containing the ID suffix text, which are assumed to be performance ID definitions. For example, the line

```
#define ADIO_M_APPMAIN_PERF_ID  0x000000D0
```

creates the ID "0x000000D0" with the name "ADIO_M_APPMAIN_PERF_ID" in the performance ID list.

Once the search is complete a dialog is displayed (see Figure 17 for an example) showing the resulting number of IDs found, the number of header files searched, and if there are any duplicate IDs. An ID is deemed a duplicate if it is defined more than once within the files of the current search. An ID is not listed as a duplicate if it matches an ID that was in the list prior to the search. In either case, the new instance is not added to the ID list.

Figure 17. Example Search Results dialog

4.6 Plot Menu

The **Plot** menu has selections for choosing which type of plot to display and for altering the plots' appearances.

4.6.1 Step

This menu item is available only if a log file is loaded. It causes the **Step** tab in the main application window to be brought to the foreground and displays the step plot (see Figure 18). Selecting the **Step** tab is equivalent to choosing the menu item. The step plot shows the entry and exit events plotted against time for individual performance IDs. The step plot is divided into multiple parts, described in detail below.
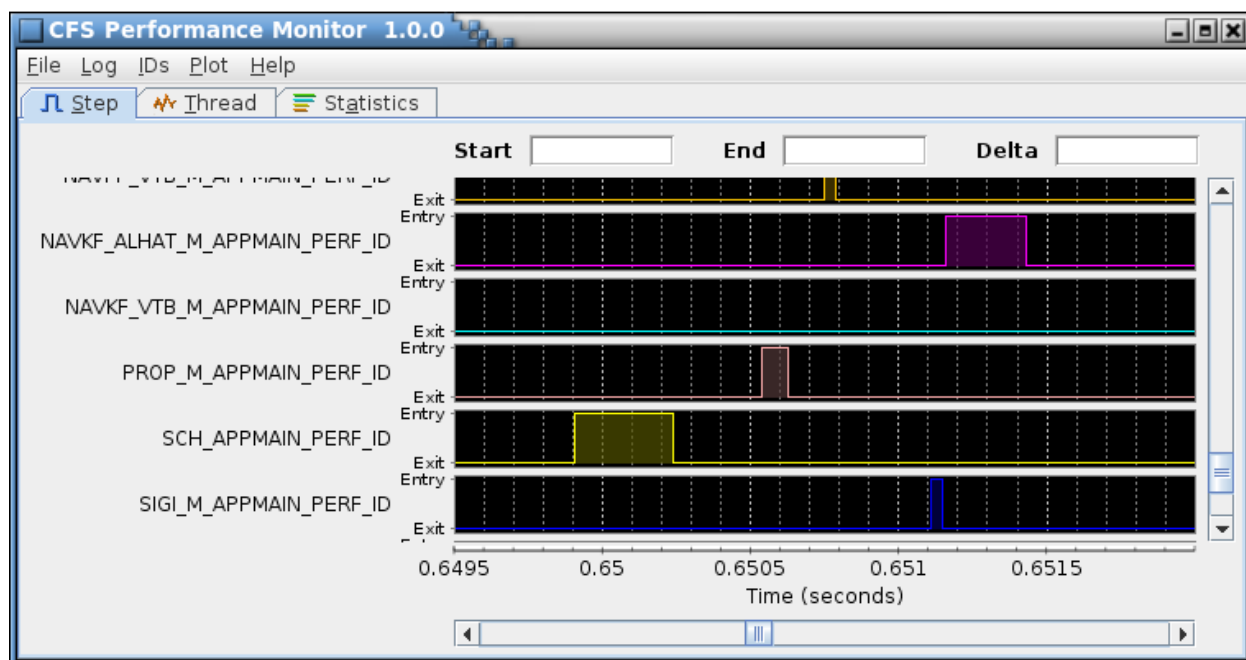
Figure 18. Step plot

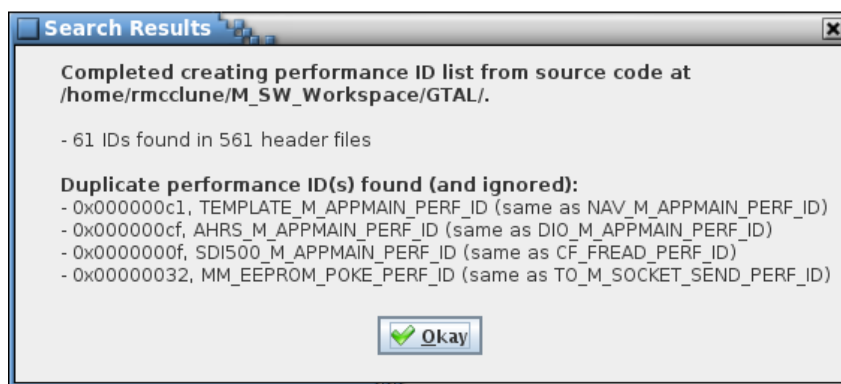The plotting area comprises the main part of the step plot display. Each ID is plotted separately and these plots are stacked vertically so that they share a common x (time) axis. An additional plot, labeled **CPU Activity**, is displayed as the topmost of the stacked plots. The CPU active status is based on the other IDs currently plotted; if any ID is active (between an entry and exit event) then the CPU is plotted as "busy"; if no IDs are active the CPU plot shows "idle." When only a single ID is selected to plot the CPU Activity plot isn't displayed. If no IDs are selected for plotting then only the CPU Activity plot is displayed, indicating idle for the duration of the log file. The CPU Activity plot line is adjusted based on the user-selected plot background color (the line color is white when the background color is dark and black when the background is light); the ID plot line colors can be adjusted using the ID editor (via the **IDs | Edit List** menu command; see paragraph 4.5.3). Colors are automatically assigned to the IDs if the user doesn't manually adjust them. The plot can be expanded by resizing the main application window.

The plot's axes can be scrolled to the left, right, up, and down by use of the mouse, mouse wheel, or keyboard. Pressing the Ctrl key, then pressing and holding the left mouse button while over the plot and dragging the pointer causes the plot to scroll left or right depending on the direction the pointer is dragged. The horizontal scroll bar located underneath the plots is also used to scroll the x-axis left or right. The vertical scroll bar located at the extreme right edge of the plots can be used to scroll the plot up or down in order to view the ID plots that don't fit within the main display window. The mouse can be used to scroll the axes by clicking the arrows at the end of the scroll bar to scroll a small amount. Clicking at either end of the scroll "thumb" causes the plot to shift a larger amount. The thumb can be selected via the mouse pointer (press and hold the left button while the pointer is over the thumb) and dragged; the plot is scrolled according to the drag direction. When the mouse pointer is positioned over a scroll bar, rotating the mouse wheel scrolls the plot based on the wheel rotation direction and amount. The keyboard can be used for scrolling via the arrow, Page Up, and Page Down keys. The up and down arrows produce the same result as clicking the up and down arrows on the vertical scroll bar. The left and right arrow keys act similarly for the horizontal scroll bar. Pressing the Shift key in conjunction with an arrow key produces the same effect as clicking at either end of the thumb (larger scroll shift). The Page Up and Page Down keys are identical to the Shift+up arrow and Shift+down arrow keys respectively. The plot must have the keyboard focus in order for the keys to produce the desired scroll actions. Click on the plot or use the Tab key to shift the focus to the plot.

When the mouse pointer is over a plot its shape changes to crosshairs and a white vertical line is drawn through the center of the crosshairs, extending across all of the IDs' plots. Above the plots are three text display boxes, labeled "Start", "End", and "Delta." As the crosshair mouse pointer moves across the plots the **Start** field displays the x-axis (time) position of the pointer. Left-clicking the mouse fixes the vertical line at the current location as well as the time in the **Start** field. Another vertical line continues to follow the pointer and its x-axis value is displayed in the **End** field. The **Delta** field now dynamically displays the difference in time between the **Start** and **End** lines. Left-clicking again fixes the second vertical line and its x-axis value in the **End** field.   The area between the **Start** and **End** lines is shaded and the Delta field shows the difference between the **End** and **Start** times. Left-clicking once again removes the vertical lines and shading, erases the three text fields, and a vertical line once again follows the crosshair pointer while updating the **Start** field. An example showing the text display boxes, vertical lines, and shading described above is shown in figure 17.
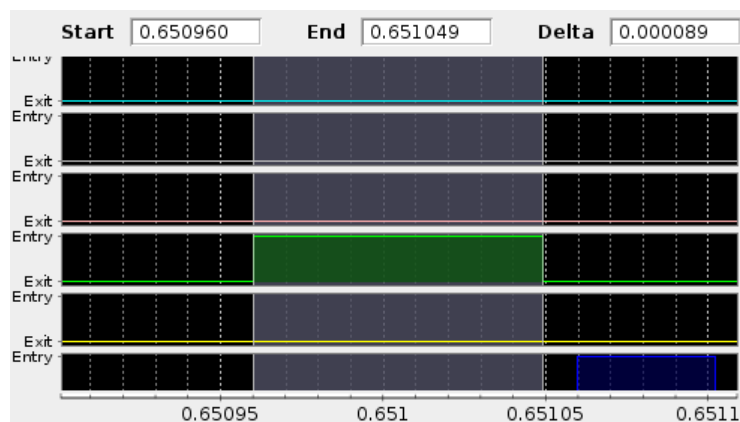

Figure 19. Start, End, and Delta markers

To quickly determine the time between two consecutive events for the same ID hold the Ctrl key when the left mouse button is pressed while the crosshair pointer is over a plot. The **Start** line is set to the nearest event to the left of the pointer and the **End** line to the nearest event to the right of the pointer. The area between is shaded and the **Start**, **End**, and **Delta** fields are updated appropriately. Any **Start** or **End** lines and shading existing prior to this are removed.

Another method to quickly set the **Start** or **End** lines is to hold the Shift key when the left mouse button is pressed while the crosshair pointer is over a plot. The currently active line (**Start** or **End**) is set to the event nearest to the pointer (to the left or right), chosen from the plot over which the crosshair pointer is placed. In this manner the difference in time between any two events, for the same or different IDs, can be accurately found.

If the mouse pointer is placed over a plotted data point for a brief period a tool tip appears showing the performance ID name, time stamp, and annotation (if any) for the logged event (see Figure 20).
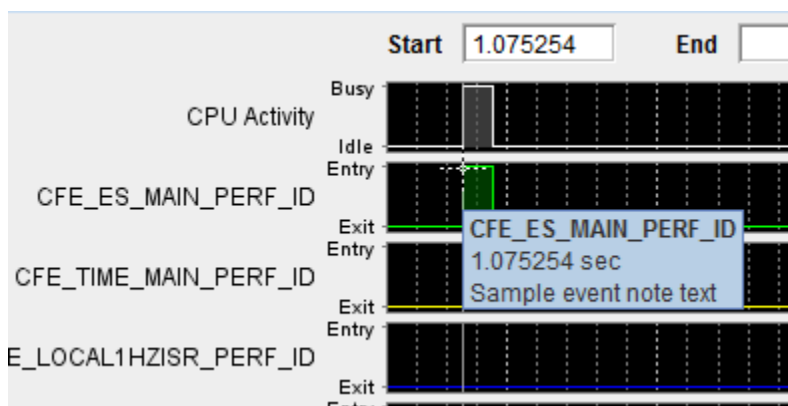
Figure 20.   Example tool tip with annotation

If a frequency value is assigned to an ID then a check is made for possible frame overruns; i.e., the actual frequency of the event represented by the ID is less than the assigned frequency, or in other words the duration of the ID's active period is greater than the inverse of the assigned frequency. Overruns are indicated on the plot by red circles around the data point where the overrun occurs. The tool tip for the data point with an overrun shows the amount of time, in seconds, that the event exceeds its allotted time. See Figure 21. If the data point also contains an annotation the text appears beneath the overrun information.

Figure 21.   Step plot frame overrun indicator

When more than one log file is loaded the plot indicates the time gap between the files using a vertical red line at the end of one file's data and the beginning of the next. The area in between is shaded as well. If the gaps are hidden via selection in the Preferences menu (see paragraph 1.0) then the shading is removed and only a single red line delineates the gap between the log files' data.

Several aspects of the plot appearance can be changed using the **Plot | Preferences** menu. These are described in detail in paragraphs 1.0 and 4.6.4.2.

The x (time) axis can be zoomed in and out using either the mouse, mouse wheel, or keyboard. Pressing the '+' (plus) key zooms the plot in. The zoom point is midway between the minimum and maximum axis values. Each press of the '+' key zooms in 5%; i.e., the x-axis range is multiplied by 0.95 to give the new range. The '-' (minus) key acts similarly, but zooms out ~5%; the x-axis range is multiplied by 1 / 0.95 to give the new range.

Rotating the mouse wheel zooms the plot in or out depending on the direction the wheel is rotated. Note that certain Microsoft wireless mice wheels do not function properly in Java under Microsoft Windows. The symptom is that the plot zooms out regardless of the direction the wheel is rotated. See 0 for steps to correct the issue if it arises.
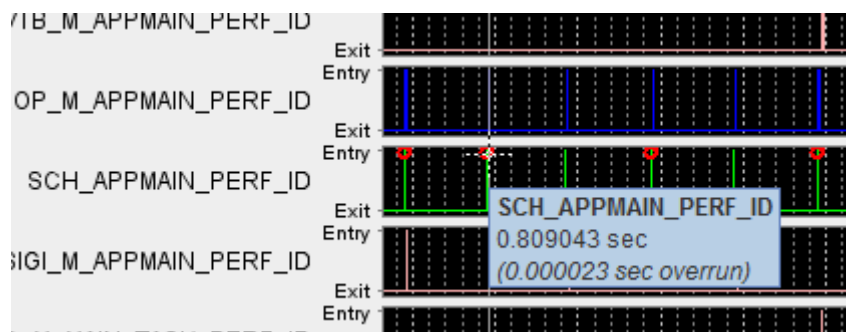
A section of the plot may be zoomed in on by positioning the mouse pointer over one of the step plots, left-clicking the mouse and dragging the pointer to the right while continuing to hold the left button down, then releasing the mouse button. While dragging the zoom area is highlighted for the plot underneath the pointer when the mouse button was first pressed. The start and end times of the zoom area become the minimum and maximum x-axis values for the plots. To restore the x-axis to the initial minimum and maximum values perform the same left-click and drag operation, but to the left.

The last method for zooming is via the context menu. The context menu appears if the mouse is right-clicked while over a plot. The context menu has options to zoom in and out in the domain (x) axis, range (y) axis, or both domain and range axes. The **Auto Range** menu can be used to restore the x-axis zoom level to the initial minimum and maximum values. Since the y-axis zoom is unchangeable for the step plots the range menu items have no effect.

In addition to the zoom commands, the context menu includes commands to copy the current plot image, save the plot image in Portable Network Graphics (png) format, and output the plot image to the printer. All of the ID step plots, not just the ones visible on screen, are output using these commands. The **Copy** command stores the image of the plots in the clipboard buffer. From there the image may be pasted into other applications by pressing Ctrl-V or by the application's **Paste** command. The **Save as** command produces a dialog asking for a file name and location to store the image file. Selecting the **Print…** command first brings up a **Page Setup** dialog that allows for arranging the image on the page; once **Okay** is pressed a second dialog appears for choosing the target printer.

4.6.2   Thread

This menu item is available only if a log file is loaded. It causes the **Thread** tab in the main application window to be brought to the foreground and displays the thread of execution plot (see Figure 22). Selecting the **Thread** tab is equivalent to choosing the menu item, though the tab is selectable even when no log is loaded. The thread of execution plot shows the entry and exit events plotted against time for individual performance IDs. Unlike the step plot, the IDs are displayed on the same plot, with each occupying a discrete row on the y-axis. A given ID's entry events are plotted as points, represented by colored symbols, just above the line delineated by the ID's label on the y-axis. The ID's exit events

are plotted using the same color and symbol just below this line. A line is drawn connecting the event points in time sequence. The thread of execution plot is divided into multiple parts, described in detail below.
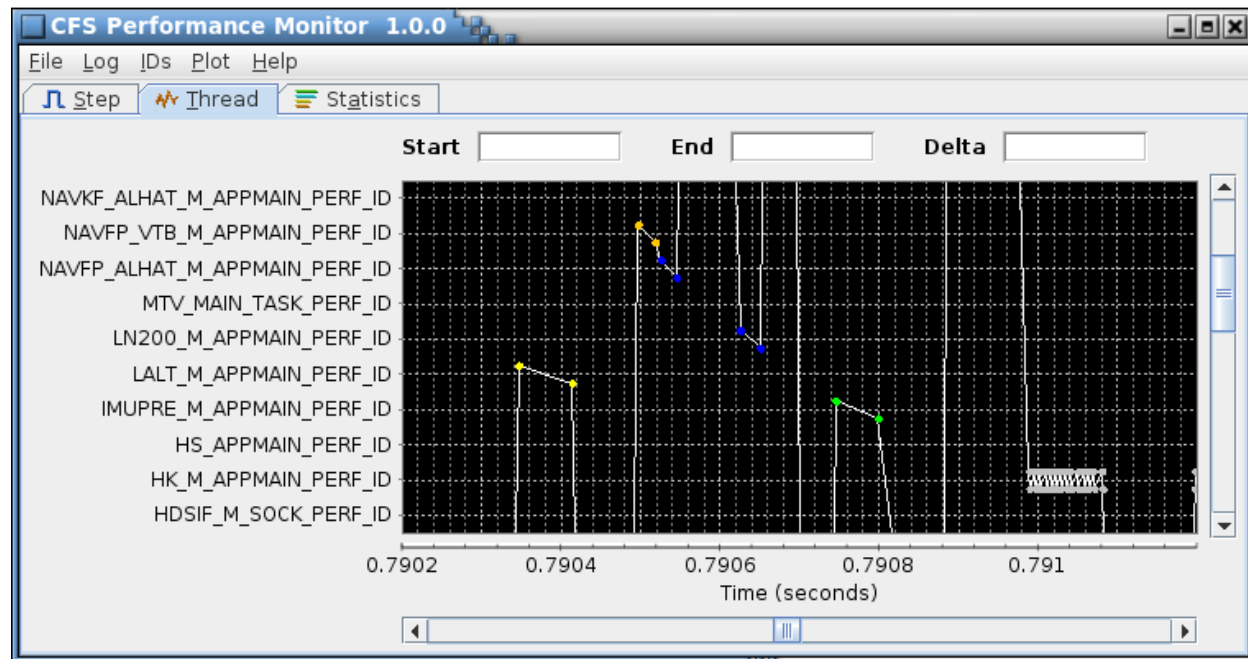

Figure 22.   Thread of execution plot

The plotting area comprises the main part of the thread of execution plot display. Every ID is plotted on the same plot so that they share a common x (time) axis. If no IDs are selected for plotting then an empty plot is displayed. The thread of execution plot line is adjusted based on the user-selected plot background color (the line color is white when the background color is dark and black when the background is light); the ID plot line colors can be adjusted using the ID editor (via the **IDs** | **Edit List** menu command; see paragraph 4.5.3). Colors are automatically assigned to the IDs if the user doesn't manually adjust them. The plot can be expanded by resizing the main application window.

The plot's axes can be scrolled to the left, right, up, and down by use of the mouse, mouse wheel, or keyboard. Pressing the Ctrl key, then pressing and holding the left mouse button while over the plot and dragging the pointer causes the plot to scroll left, right, up, or down depending on the direction the pointer is dragged. The horizontal scroll bar located underneath the plot is also used to scroll the x-axis left or right. The vertical scroll bar located at the extreme right edge of the plot can be used to scroll the plot up or down in order to view the IDs that don't fit within the main display window. The mouse can be used to scroll the axes by clicking the arrows at the end of the scroll bar to scroll a small amount. Clicking within the scroll bar to either end of the scroll "thumb" causes the plot to shift a larger amount. The thumb can be selected via the mouse pointer (press and hold the left button while the pointer is over the thumb) and dragged; the plot is scrolled according to the drag direction. When the mouse pointer is positioned over a scroll bar, rotating the mouse wheel scrolls the plot based on the wheel rotation direction and amount. The keyboard can be used for scrolling via the arrow, Page Up, and Page Down keys. The up and down arrows produce the same result as clicking the up and down arrows on the vertical scroll bar. The left and right arrow keys act similarly for the horizontal scroll bar. Pressing

the Shift key in conjunction with an arrow key produces the same effect as clicking at either end of the thumb (larger scroll shift). The Page Up and Page Down keys are identical to the Shift+up arrow and Shift+down arrow keys respectively. The plot must have the keyboard focus in order for the keys to produce the desired scroll actions. Click on the plot or use the Tab key to shift the focus to the plot.

When the mouse pointer is over the plots its shape changes to crosshairs and a white vertical line is drawn through the center of the crosshairs, extending across the entire plot. Above the plots are three text display boxes, labeled "Start", "End", and "Delta." As the crosshair mouse pointer moves across the plots the **Start** field displays the x-axis (time) position of the pointer. Left-clicking the mouse fixes the vertical line at the current location as well as the time in the **Start** field. Another vertical line continues to follow the pointer and its x-axis value is displayed in the **End** field. The **Delta** field now dynamically displays the difference in time between the **Start** and **End** lines. Left-clicking again fixes the second vertical line and its x-axis value in the **End** field. The area between the **Start** and **End** lines is shaded and the Delta field shows the difference between the **End** and **Start** times. Left-clicking once again removes the vertical lines and shading, erases the three text fields, and a vertical line once again follows the crosshair pointer while updating the **Start** field.

To quickly determine the time between two consecutive events for the same ID hold the Ctrl key when the left mouse button is pressed while the crosshair pointer is over a plot. The **Start** line is set to the nearest event to the left of the pointer and the **End** line to the nearest event to the right of the pointer. The area between is shaded and the **Start**, **End**, and **Delta** fields are updated appropriately. Any **Start** or **End** lines and shading existing prior to this are removed.

Another method to quickly set the **Start** or **End** lines is to hold the Shift key when the left mouse button is pressed while the crosshair pointer is over a plot. The currently active line (**Start** or **End**) is set to the event nearest to the pointer (to the left or right), chosen from the plot over which the crosshair pointer is placed. In this manner the difference in time between any two events, for the same or different IDs, can be accurately found.

If the mouse pointer is placed over a plotted data point for a brief period a tool tip appears showing the performance ID name, time stamp, and annotation (if any) for the logged event (similar to that seen in Figure 20).

If a frequency value is assigned to an ID then a check is made for possible frame overruns; i.e., the actual frequency of the event represented by the ID is less than the assigned frequency, or in other words the duration of the ID's active period is greater than the inverse of the assigned frequency. Overruns are indicated on the plot by red circles around the data point where the overrun occurs (see Figure 23). The tool tip for the data point with an overrun shows the amount of time, in seconds, that the event exceeds its allotted time, identical to that for the step plot as shown in Figure 21. If the data point also contains an annotation the text appears beneath the overrun information.
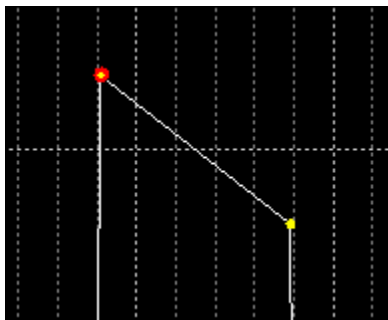
Figure 23. Thread of execution plot frame overrun indicator

When more than one log file is loaded the plot indicates the time gap between the files using a vertical red line at the end of one file's data and the beginning of the next. The area in between is shaded as well. If the gaps are hidden via selection in the Preferences menu (see paragraph 1.0) then the shading is removed and only a single red line delineates the gap between the log files' data.

Several aspects of the plot appearance can be changed using the **Plot | Preferences** menu. These are described in detail in paragraphs 1.0 and 4.6.4.2.

The x (time) and y (ID) axes can be zoomed in and out using either the mouse, mouse wheel, or keyboard. Pressing the '+' (plus) key zooms the plot in both axes. The zoom point is midway between the minimum and maximum axis values. Each press of the '+' key zooms in 5%; i.e., the x-axis range is multiplied by 0.95 to give the new range. The '-' (minus) key acts similarly, but zooms out ~5%; the x-axis range is multiplied by 1 / 0.95 to give the new range.

Rotating the mouse wheel zooms the plot in or out depending on the direction the wheel is rotated. Note that certain Microsoft wireless mice wheels do not function properly in Java under Microsoft Windows. The symptom is that the plot zooms out regardless of the direction the wheel is rotated. See 0 for steps to correct the issue if it arises.

A section of the plot may be zoomed in on by positioning the mouse pointer over the thread of execution plot, left-clicking the mouse and dragging the pointer to the right and up or down while continuing to hold the left button down, then releasing the mouse button. While dragging the zoom area is highlighted for the plot underneath the pointer when the mouse button was first pressed. The start and end times of the zoom area become the minimum and maximum x-axis values for the plot, and the lower and upper IDs within the zoom area become the minimum and maximum y-axis values. To restore the axes to the initial minimum and maximum values perform the same left-click and drag operation, but to the left.

The last method for zooming is via the context menu. The context menu appears if the mouse is right-clicked while over a plot. The context menu has options to zoom in and out in the domain (x) axis, range (y) axis, or both domain and range axes. The **Auto Range** menu can be used to restore the x-axis zoom level to the initial minimum and maximum values.

In addition to the zoom commands, the context menu includes commands to copy the current plot image, save the plot image in Portable Network Graphics (png) format, and output the plot image to the printer. The **Copy** command stores the image of the plot in the clipboard buffer. From there the image may be pasted into other applications by pressing Ctrl-V or by the application's **Paste** command. The **Save as** command produces a dialog asking for a file name and location to store the image file. Selecting

the **Print…** command first brings up a **Page Setup** dialog that allows for arranging the image on the page; once **Okay** is pressed a second dialog appears for choosing the target printer.

4.6.3   Statistics

Selecting the Statistics menu item causes a submenu to appear that has the various statistics plots to choose from. Selecting the Statistics tab also displays one of the statistics plots – the plot initially shown is ID Time Active. The plots are in the form of pie and bar charts, with each pie slice or bar representing a performance ID (see Figure 24 and Figure 25). The size of the slice or length of the bar indicates the relative percentage that this ID accounts for of the selected statistic. For the pie charts the ID names surround the chart with lines connecting them to their respective pie slices. The plot can be expanded by resizing the main application window.
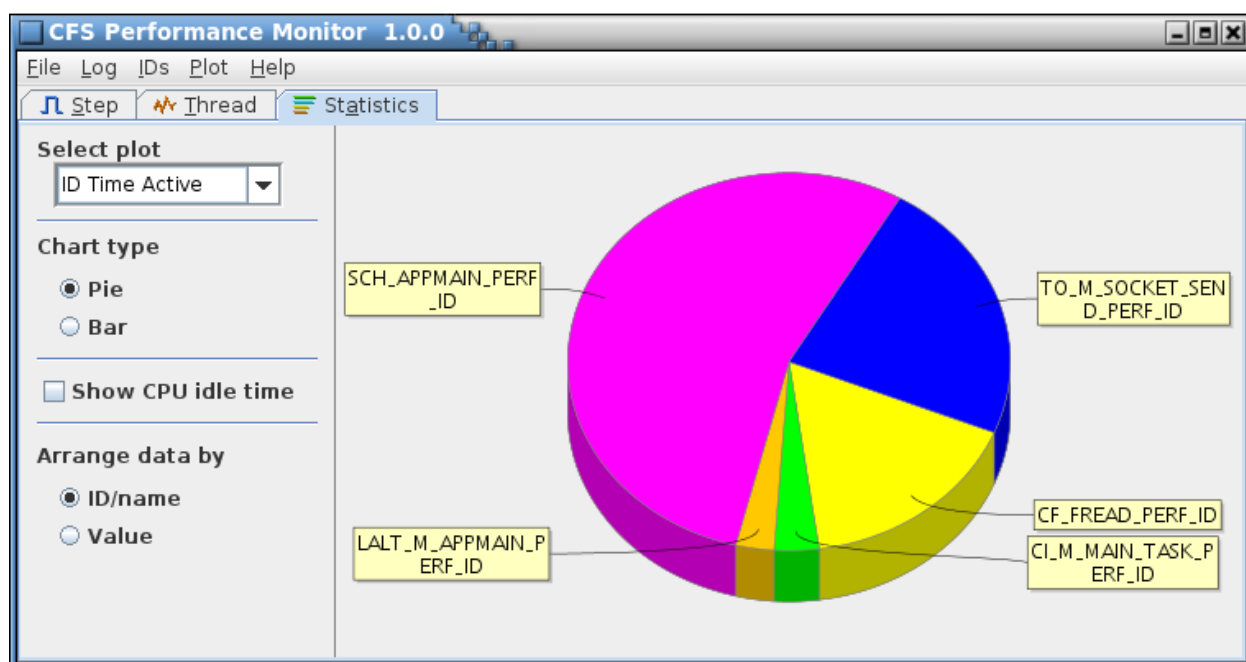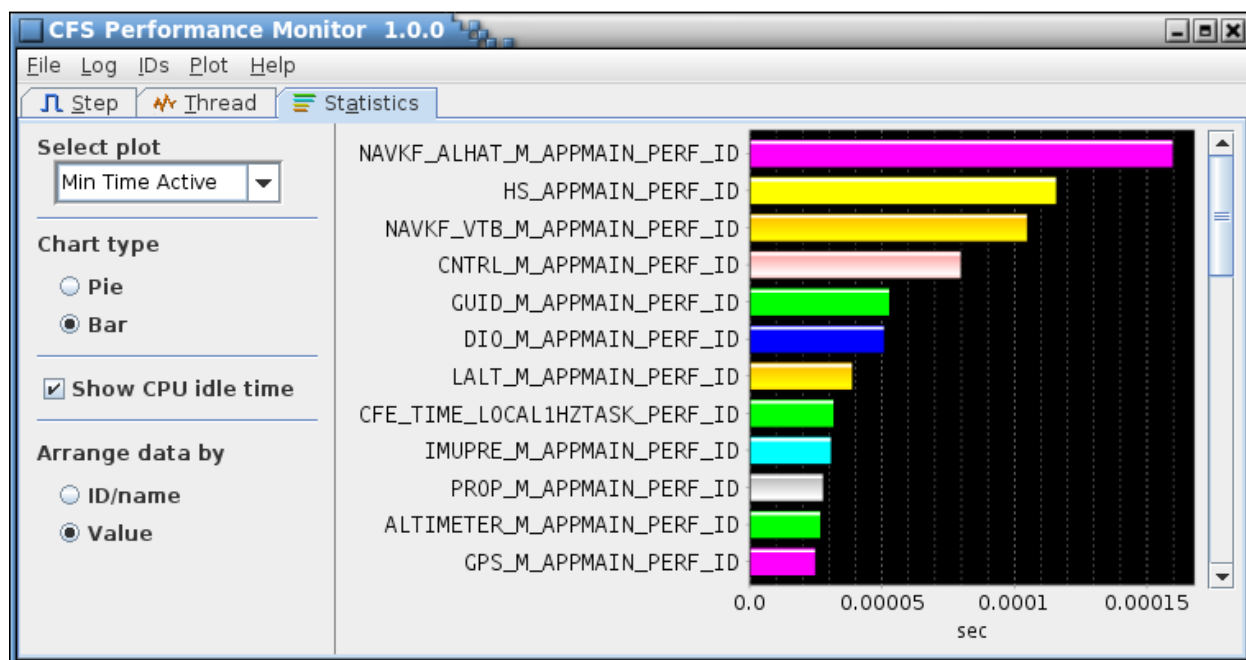

Figure 24.   Statistics pie chart

Figure 25. Statistics bar chart

The chart ID colors can be adjusted using the ID editor (via the **IDs** | **Edit List** menu command; see paragraph 4.5.3). Colors are automatically assigned to the IDs if the user doesn't manually adjust them.

The pie chart can be rotated clockwise or counter-clockwise by rotating the mouse wheel. Pressing the plus (+) or minus (-) keys produces the identical effect. The ID labels follow this motion, which can make it easier to locate a particular ID. Positioning the mouse pointer over a slice and pausing causes a tool tip to appear containing the ID name (or ID is no name is assigned), the statistic's value and units, and the percentage of the plot this value represents.

Rotating the mouse wheel when the mouse pointer is positioned over a bar chart zooms the chart's x-axis in and out, based on the rotation direction. The minimum x value remains fixed at zero. When zooming in this manner the maximum x-axis value is constrained such that is won't fall below the precision of the log data or go beyond the maximum statistical value displayed in the chart. The plus (+) and minus (-) keys act to zoom the chart in and out in an identical manner. Zooming can also be performed by positioning the mouse pointer over the plot, pressing the left mouse button, and dragging the pointer to the right; the x-axis maximum value is changed according to the mouse pointer's ending position (the minimum remains anchored at 0.0). Pressing the button and dragging to the left restores the plot to its original x-axis values. Positioning the mouse pointer over a bar and pausing causes a tool tip to appear containing the ID name (or ID is no name is assigned), and the statistic's value and units.

The bar chart's y-axis can be scrolled up and down by use of the mouse, mouse wheel, or keyboard. Pressing the Ctrl key, then pressing and holding the left mouse button while over the chart and dragging the pointer causes the chart to scroll left, right, up, or down depending on the direction the pointer is dragged. The vertical scroll bar located at the extreme right edge of the chart can be used to view the IDs that don't fit within the main display window. The mouse can be used to scroll the y-axis by clicking the arrows at the end of the scroll bar to scroll a small amount. Clicking to either side of the scroll "thumb" causes the chart to shift a larger amount. The thumb can be selected via the mouse pointer

(press and hold the left button while the pointer is over the thumb) and dragged; the chart is scrolled according to the drag direction. When the mouse pointer is positioned over the y-axis scroll bar, rotating the mouse wheel scrolls the chart based on the wheel rotation direction and amount. The keyboard can be used for scrolling via the arrow, Page Up, and Page Down keys. The up and down arrows produce the same result as clicking the up and down arrows on the vertical scroll bar. Pressing the Shift key in conjunction with an arrow key produces the same effect as clicking to either side of the thumb (larger scroll shift). The Page Up and Page Down keys are identical to the Shift+up arrow and Shift+down arrow keys respectively. The chart must have the keyboard focus in order for the keys to produce the desired scroll actions. Click on the chart or use the Tab key to shift the focus to the chart.

Right-clicking the chart produces a popup context menu. This menu includes commands to copy the current chart image, save the chart image in Portable Network Graphics (png) format, and output the chart image to the printer. The **Copy** command stores the image of the chart in the clipboard buffer. From there the image may be pasted into other applications by pressing Ctrl-V or by the application's **Paste** command. The **Save as** command produces a dialog asking for a file name and location to store the image file. Selecting the **Print…** command first brings up a **Page Setup** dialog that allows for arranging the image on the page; once **Okay** is pressed a second dialog appears for choosing the target printer.

To the left of the chart are controls for selecting the statistic to plot and for altering how the data is displayed. These controls are as follows:

Select plot   The statistic currently plotted is shown in the combo box. Left-clicking the arrow on the right of the combo box causes a dropdown menu to appear (see Figure 26) that lists the other statistics values. The mouse (left-click) or keyboard (arrows and space bar) are used to select one of the statistics. The pie chart is automatically updated to display the new values. Descriptions of the individual statistics charts is given in paragraphs 4.6.3.1 through 4.6.3.9.
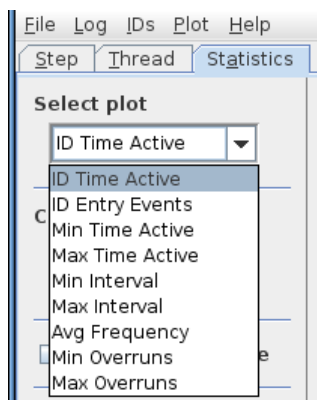
Figure 26. Statistics plot selections

Chart type Pie   This radio button is one of a mutually exclusive pair. When **Pie** is selected the statistics values represented by the **Selected plot** combo box are displayed in a pie chart. The default is **Pie**. This option is also available in the **Plot | Preferences** menu under the **Statistics** tab (see Figure 30), and selection/deselection in one location is reflected in the other

Chart type Bar This radio button is one of a mutually exclusive pair. When **Bar** is selected the statistics values represented by the **Selected plot** combo box are displayed in a bar chart. The default is **Pie**. This option is also available in the **Plot** | **Preferences** menu under the **Statistics** tab (see Figure 30), and selection/deselection in one location is reflected in the other

Show CPU idle time The CPU idle time is defined as the accumulated portions of time during which none of the IDs selected for display is active (between entry and exit events). When this box is checked the idle time is shown in the pie charts (the bar charts do not indicate CPU idle time regardless of the check box setting). If unchecked the idle time is not counted and only the IDs are plotted. The default is checked. This option is also available in the **Plot** | **Preferences** menu under the **Statistics** tab (see Figure 30), and selection/deselection in one location is reflected in the other

Arrange data by ID/name This radio button is one of a mutually exclusive pair. When **ID/name** is selected the IDs in the chart are ordered alphabetically by performance ID name, or, if no name is associated with the ID, then by the ID itself. The default is **ID/name**. This option is also available in the **Plot** | **Preferences** menu under the **Statistics** tab (see Figure 30), and selection/deselection in one location is reflected in the other

Arrange data by Value This radio button is one of a mutually exclusive pair. When **Value** is selected the IDs in the pie chart are ordered based on the value of the statistic being plotted (e.g., Min Time Active). The default is **ID/name**. This option is also available in the **Plot** | **Preferences** menu under the **Statistics** tab (see Figure 30), and selection/deselection in one location is reflected in the other

4.6.3.1 ID Time Active

This menu item displays the ID Time Active statistics chart. Each slice/bar of this chart represents the time that a specific ID is active (i.e., between entry and exit events). In the pie chart format the size of the slice is proportional to the total of all IDs' active times, so IDs with larger slices are active for a greater amount of time than those with smaller slices. In general, for the pie chart the idle time (the total time when no ID is active) dominates this plot. Unchecking the **Show CPU idle time** check box removes the idle time from the chart, making it easier to see the contribution of the IDs.

4.6.3.2 ID Entry Events

This menu item displays the ID Entry Events statistics chart. Each slice/bar of this chart represents the number of entry events that occurred for a specific ID. In the pie chart format the size of the slice is proportional to the total of all IDs' entry events, so IDs with larger slices experienced a greater number of entry events than those with smaller slices.

4.6.3.3 Avg Frequency

This menu item displays the Avg Frequency statistics chart. Each slice/bar of this chart represents the frequency at which entry events occur for a specific ID, averaged over the duration of the log file (e.g., if 3 entry events are logged and the file duration is 2 seconds then the average frequency is: 3 events ÷ 2 seconds = 1.5 events/second). In the pie chart format the size of the slice is proportional to the total of

all IDs' average frequencies, so entry events for IDs with larger slices occurred with greater frequency than those with smaller slices.

4.6.3.4   Min Time Active

This menu item displays the Min Time Active statistics chart. Each slice/bar of this chart represents the shortest amount of time a specific ID was active (i.e., between entry and exit events) based on all of that ID's active periods. In the pie chart format the size of the slice is proportional to the total of all IDs' minimum active times, so IDs with smaller slices had shorter active durations than those with larger slices.

4.6.3.5   Max Time Active

This menu item displays the Max Time Active statistics chart. Each slice/bar of this chart represents the longest amount of time a specific ID was active (i.e., between entry and exit events) based on all of that ID's active periods. In the pie chart format the size of the slice is proportional to the total of all IDs' maximum active times, so IDs with larger slices had longer active durations than those with smaller slices.

4.6.3.6   Min Interval

This menu item displays the Min Interval statistics chart. Each slice/bar of this chart represents the shortest amount of time between consecutive entry events for a specific ID. In the pie chart format the size of the slice is proportional to the total of all IDs' minimum interval times, so IDs with smaller slices had shorter minimum interval durations than those with larger slices.

4.6.3.7   Max Interval

This menu item displays the Max Interval statistics chart. Each slice/bar of this chart represents the longest amount of time between consecutive entry events for a specific ID. In the pie chart format the size of the slice is proportional to the total of all IDs' maximum interval times, so IDs with larger slices had longer maximum interval durations than those with smaller slices.

4.6.3.8   Min Overruns

This menu item displays the Min Overruns statistics chart. Each slice/bar of this chart represents the shortest amount of time a specific ID overran its allotted time (the allowed time is the inverse of the ID frequency) based on all of that ID's overruns. In the pie chart format the size of the slice is proportional to the total of all IDs' minimum overrun times, so IDs with smaller slices had a shorter overrun duration than those with larger slices.

4.6.3.9   Max Overruns

This menu item displays the Max Overruns statistics chart. Each slice/bar of this chart represents the longest amount of time a specific ID overran its allotted time (the allowed time is the inverse of the ID frequency) based on all of that ID's overruns. In the pie chart format the size of the slice is proportional to the total of all IDs' maximum overrun times, so IDs with larger slices had a longer overrun duration than those with smaller slices.

4.6.4   Preferences

The plot Preferences dialog enables altering the appearance of the various plot types (step, thread of execution, and statistics). The dialog is divided into four tabs. Any changes to the preference selections are implemented in the plots when the **Okay** button is pressed. Also, the values are saved in the

program preferences location so that the preference values are retained for the next time the application is opened. Pressing the **Default** button restores all of the preferences to the default values (shown in Table 2). Pressing the **Cancel** button exits the dialog without changing the plot preferences. The Plot Preferences dialog controls can be activated using the mouse, mouse wheel (for the slider controls), or keyboard.

4.6.4.1   All

The first preferences tab is the **All** tab (Figure 27), which is for settings that affect more than one plot type.
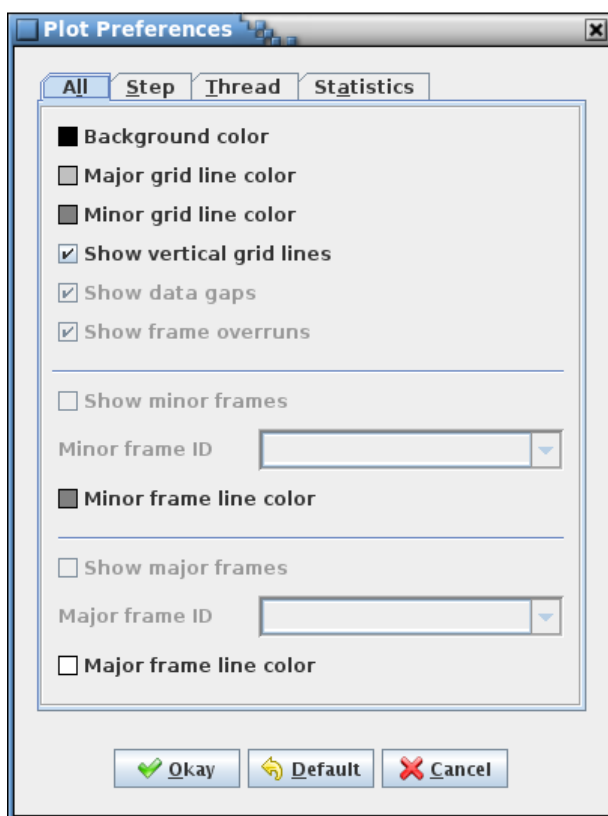
Figure 27.   Plot Preferences dialog – All tab

Background color   The small colored button to the left of the label represents the currently selected color for the plot background. Pressing the button causes a color choice dialog to appear (see Figure 15). After selecting a new color in the color choice dialog press its **Okay** button to return to the Preferences dialog. The button color changes to show the new choice. The default is black

Major grid line color   The small colored button to the left of the label represents the currently selected color for the plot major grid lines. Pressing the button causes a color choice dialog to appear (see Figure 15). After selecting a new color in the color choice dialog press its **Okay** button to return to the Preferences dialog. The button color changes to show the new choice. The default is light gray

Minor grid line color   The small colored button to the left of the label represents the currently selected color for the plot minor grid lines. Pressing the button causes a color choice dialog to appear (see Figure 15). After selecting a new color in the color choice dialog press its **Okay** button to return to the Preferences dialog. The button color changes to show the new choice. The default is gray

Show vertical grid lines   If checked the step and thread of execution plots display vertical grid lines. If unchecked the grid lines are not drawn for these plots. The default is checked

Show data gaps   This option is available only while multiple logs are loaded, otherwise it is grayed out. When checked the gap in time between each log is plotted in the step and thread of execution plots. If unchecked the gaps are removed and the logged data appears contiguous. In either case vertical red lines on the plot indicate where the gaps start and stop. The default is checked

Show frame overruns   This option is available only if one or more IDs has a non-zero frequency assigned. When checked any frame overruns are indicated on the step and thread of execution plots. If unchecked the indicators are not displayed.

Show minor frames   Checking this box causes vertical lines in the color indicated by the **Minor frame line color** to be drawn across the step and thread of execution plots at each entry event for the ID selected in the **Minor frame ID** dropdown menu. The default is unchecked

Minor frame ID   This drop down menu provides a list of the currently loaded IDs in alphanumeric order. The ID names are displayed, or, if no name has been defined for an ID, the ID value is displayed instead. The ID chosen should represent the event marking the beginning of the CFS minor frame; however, any ID can be chosen. The **Show minor frames** check box must be enabled for the minor frame vertical lines to be displayed on the plots. The default is blank (no ID selected)

Minor frame line color   The small colored button to the left of the label represents the currently selected color for the minor frame lines. Pressing the button causes a color choice dialog to appear (see Figure 15). After selecting a new color in the color choice dialog press its **Okay** button to return to the Preferences dialog. The button color changes to show the new choice. The default is gray

Show major frames   Checking this box causes vertical lines in the color indicated by the **Major frame line color** to be drawn across the step and thread of execution plots at each entry event for the ID selected in the **Major frame ID** dropdown menu. The default is unchecked

Major frame ID   This drop down menu provides a list of the currently loaded IDs in alphanumeric order. The ID names are displayed, or, if no name has been defined for an ID, the ID value is displayed instead. The ID chosen should represent the event marking the beginning of the CFS major frame; however, any ID can be chosen. The **Show major frames** check box must be enabled for the major frame vertical lines to be displayed on the plots. The default is blank (no ID selected)

Major frame line color   The small colored button to the left of the label represents the currently selected color for the major frame lines. Pressing the button causes a color choice dialog to appear (see Figure 15). After selecting a new color in the color choice dialog press its **Okay** button to return to the Preferences dialog. The button color changes to show the new choice. The default is white

4.6.4.2   **Step**

The second tab is the **Step** tab (Figure 28). The selections on this tab affect only the step plots.
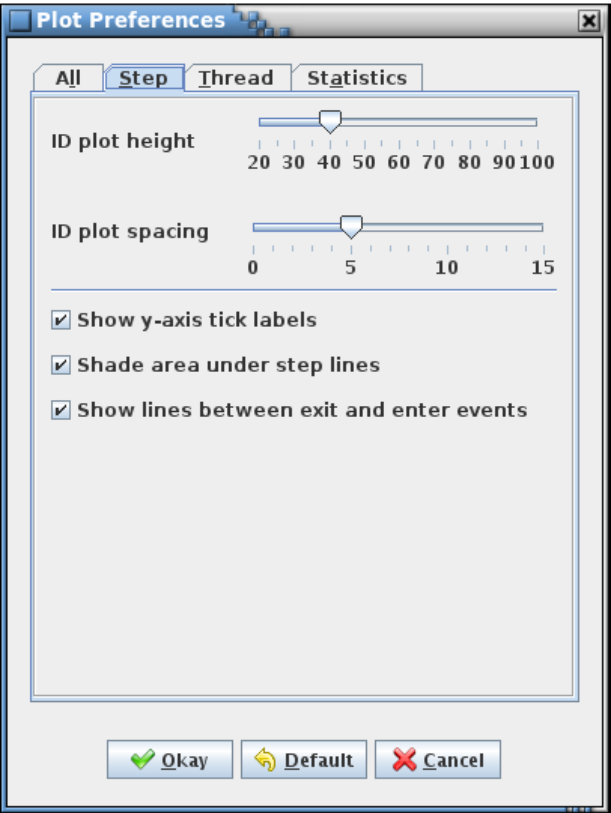
Figure 28.   Plot Preferences dialog – Step tab

Sub-plot height   This slider control adjusts the height, in pixels, of the individual ID plots that comprise the step plot. The default is 40 pixels

Sub-plot spacing   This slider control adjusts the height, in pixels, of the individual ID plots that comprise the step plot. The default is 5 pixels

Show y-axis tick labels   When checked the labels are displayed for the y-axis tick marks ("Entry" and "Exit" for the IDs, and "Busy" and "Idle" for CPU Activity). If unchecked these labels are removed. This is useful if the plot height or spacing is reduced such that the labels overwrite each other. The default is checked

Shade area under step lines   This check box determines whether or not the area underneath the step plot lines between entry and exit events is shaded. Shading makes it

easier to see when an ID is active (between entry and exit events). An example of the shading can be seen in Figure 18. Check the box to enable shading and uncheck it to remove the shading (the area is then drawn in the normal plot background color). The default is checked

Show lines between exit and enter events When this box is checked the points between exit and enter events are drawn on the step plot; i.e., the horizontal lines level with the 'Exit' y-axis tick label. Unchecking this box causes these lines to not be drawn. The horizontal lines between enter and exit events (those level with the 'Enter' y-axis tick label) are drawn regardless. Hiding the exit lines reduces clutter on the plot, particularly when the sub-plot height is shortened. The default is checked

4.6.4.3   Thread

The next tab, labeled **Thread** (Figure 29), is for changing the appearance of the thread of execution plot.
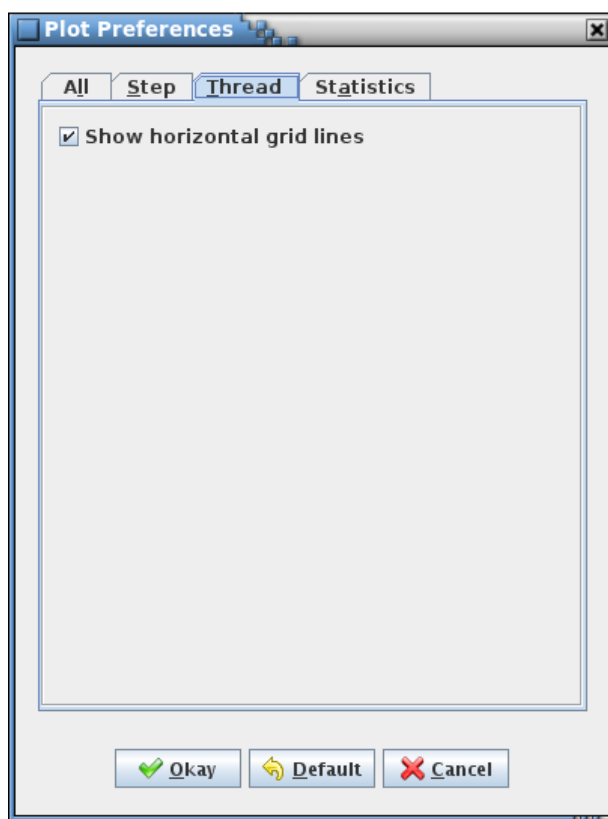

Figure 29.   Plot Preferences dialog - Thread tab

Show horizontal grid lines When checked the horizontal grid lines are displayed on the plot. When unchecked these lines are removed. The default is checked

4.6.4.4 **Statistics**

The last tab, **Statistics** (Figure 30), shows preference selections for the statistics pie charts.
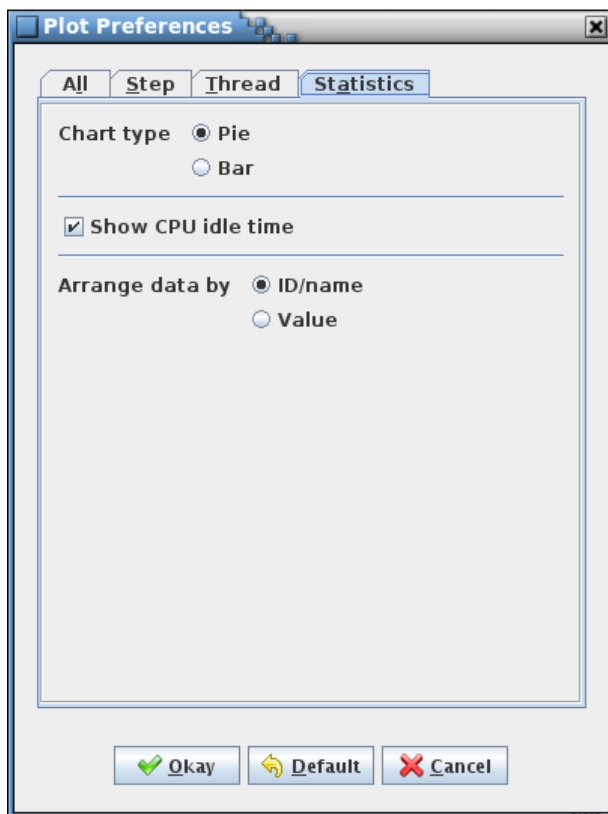
Figure 30. Plot Preferences dialog – Statistics tab

Chart type Pie	This radio button is one of a mutually exclusive pair. When **Pie** is selected the statistics values represented by the Selected plot combo box are displayed in a pie chart. The default is **Pie**. This option is also available in the Statistics plot itself (see Figure 24 or Figure 25), and selection/deselection in one location is reflected in the other
Chart type Bar	This radio button is one of a mutually exclusive pair. When **Bar** is selected the statistics values represented by the Selected plot combo box are displayed in a bar chart. The default is **Pie**. This option is also available in the Statistics plot itself (see Figure 24 or Figure 25), and selection/deselection in one location is reflected in the other
Show CPU idle time	The CPU idle time is defined as the accumulated portions of time during which none of the IDs selected for display is active (between entry and exit events). When this box is checked the idle time is shown in the pie charts (the bar charts do not indicate CPU idle time regardless of the check box setting). If unchecked the idle time is not counted and only the IDs are plotted. The default is checked. This option is also available in the Statistics

plot itself (see Figure 24), and selection/deselection in one location is reflected in the other

Arrange data by ID/name   This radio button is one of a mutually exclusive pair. When **ID/name** is selected the IDs in the pie chart are ordered alphabetically by performance ID name, or, if no name is associated with the ID, then by the ID itself. The default is **ID/name**. This option is also available in the Statistics plot itself (see Figure 24 or Figure 25), and selection/deselection in one location is reflected in the other

Arrange data by Value   This radio button is one of a mutually exclusive pair. When **Value** is selected the IDs in the pie chart are ordered based on the value of the statistic being plotted (e.g., Min Time Active). The default is **ID/name**. This option is also available in the Statistics plot itself (see Figure 24 or Figure 25), and selection/deselection in one location is reflected in the other

4.6.5   Set Bounds

The step and thread of execution plots allowing zooming in (narrowing the range of an axis in order to focus on a smaller selection of the data) and zooming out (expanding the range of axis to display more of the data in the same amount of space). Zooming is usually accomplished using the mouse; however, sometimes it's necessary to return quickly to a specific location in the data. The **Set Bounds** dialog allows for directly setting the range of the data displayed.

4.6.5.1   Step

The step plot allows zooming only in the x (time) axis. If the **Step Bounds** dialog is displayed while the step plot is visible only the x-axis values are shown (see Figure 31). When the dialog appears the minimum and maximum values reflect those currently in use in the step plot.

Change the minimum and/or maximum values and press **Okay** to update the plot to the new time range. If the minimum x value (time) is equal to or exceeds the maximum an error dialog appears indicating the problem. Press **Okay** to exit the error dialog and return to the **Set Bounds** dialog to correct the error.

 Selecting **Cancel** exits the **Set Bounds** dialog without affecting the x-axis values.
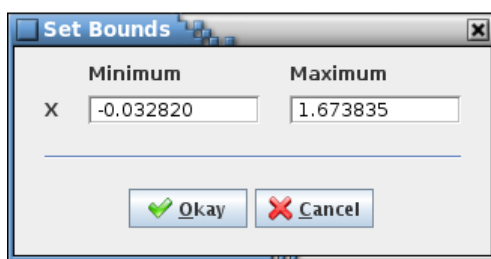
Figure 31.   Set Bounds dialog – x-axis only

4.6.5.2   Thread of Execution

The thread of execution plot allows zooming only in both the x (time) and y (ID name) axis. If the **Step Bounds** dialog is displayed while the thread of execution plot is visible both the x- and y-axes values are shown (see Figure 32). When the dialog appears the minimum and maximum values reflect those currently in use in the thread of execution plot.

For the y-axis (ID name), dropdown menus are provided for selecting from the list of available ID names (if an ID has no name assigned then its ID is used instead). The ordering of the ID names is determined by the order shown in the plot, which comes from the order chosen using the ID editor dialog (see paragraph 4.5.3 for details). The selection of a minimum ID name limits the ID names available in the maximum dropdown menu, and vice versa – the minimum ID name cannot be "less" than the maximum. The ID names for the minimum and maximum can be the same – the result is that the plot is zoomed so as to show only the single ID name.

Once the minimum and/or maximum values have been changed press **Okay** to update the plot to the new time and/or ID name range. If the minimum x value (time) is equal to or exceeds the maximum an error dialog appears indicating the problem. Press **Okay** to exit the error dialog and return to the **Set Bounds** dialog to correct the error.

Selecting **Cancel** exits the **Set Bounds** dialog without affecting the axes values.
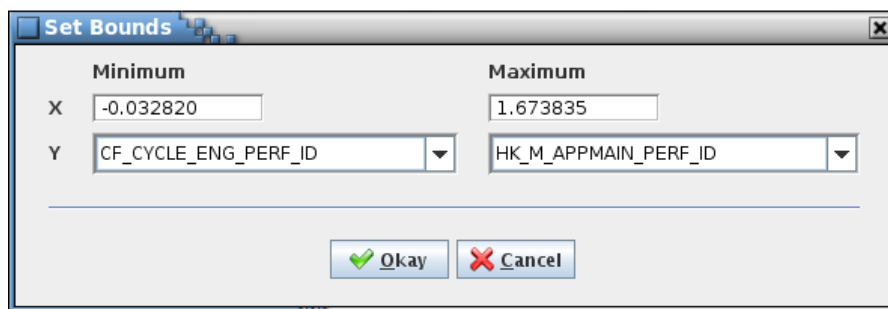
Figure 32. Set Bounds dialog – x- and y-axes

4.7 Help Menu

4.7.1 Guide

The **Guide** command displays a copy of this user's guide in PDF format.

4.7.2 About

Selecting the **About** menu item brings up an informational dialog (see Figure 33) providing the application's version information.

Figure 33. About dialog

Appendix A. Acronyms

cFE   Core Flight Executive

CFS   Core Flight Software

CPM   CFS Performance Monitor

CPU   Central Processing Unit

GUI   Graphical User Interface

I/O   Input/Output

ID   Identifier

JAR   Java Archive

JRE   Java™ Runtime Environment

L&F   Look and Feel

PDF   Portable Document Format

PNG   Portable Network Graphics

Appendix B. Definitions

Active   An ID is considered active during the time period between entry and exit events

Active Period   The interval between a performance ID's entry event and its next exit event

Domain Axis   The plot x-axis (time, in seconds)

Entry Event   The event logged when a function or section of code in CFS is entered, created by a call to the CFS function CFE_ES_PerfLogEntry(*ID*). An ID's entry events are designated in the log by having the ID's high bit cleared (= 0)

Exit Event   The event logged when a function or section of code in CFS is exited, created by a call to the CFS function CFE_ES_PerfLogExit(*ID*). An ID's exit events are designated in the log by having the ID's high bit set (= 1)

Frame Overrun   The condition when the time between a performance ID's consecutive entry events, or the time between an ID's entry event and the next exit event, exceeds an allotted amount of time

Frequency   Value assigned by the user to a performance ID for the expected entry event rate in entry events per second. The frequency is used to detect possible frame overruns

Interval   The time between a performance ID's consecutive entry events

Overrun   See Frame Overrun

Inactive   An ID is considered inactive during the time period between exit and entry events

Left-click   Pressing the left mouse button after positioning the mouse pointer over a control

Range Axis   The plot y-axis (ID names, or IDs if no name is associated with an ID)

Right-click   Pressing the right mouse button after positioning the mouse pointer over a control

Step Plot   The plot of the each selected performance ID's entry and exit events. The ID's plots are stacked vertically so as to share a common x (time stamp) axis. The entry events are shown higher on the y-axis than the exit events, and the consecutive events are connected by lines in a step-wise manner

Thread of Execution Plot   The plot of all selected performance IDs' entry and exit events. Each ID is represented by a horizontal line on the y-axis, with its entry events plotted slightly above the line and its exit events slightly below. A line directly connects each consecutive event across all of the selected IDs thereby visually showing the sequence of the events

Appendix C. Error & Warning Messages

The table below lists all of the error and warning messages, in alphabetical order, that can occur in the CPM application and the causes. An error message implies that the intended operation cannot be successfully completed. A command line error message results in immediate program termination, but for other errors the application continues to run. A warning message indicates that the though the operation was unsuccessful, program behavior is (generally) unaffected.

Type	Message	Cause
Warning	# performance IDs loaded <(*addedIDs* added)> - *badIDs* ID value(s) exceed the maximum supported (*maxID*)	This message can only occur if a log is loaded. *badIDs* number of IDs loaded from an ID file have a value that exceeds the maximum ID value supported by the current log. The maximum ID value supported, *maxID*, is equal to the filter/trigger mask size (reported in the log header) multiplied by 32, minus 1. If multiple logs are loaded which have different mask sizes then the largest of the mask sizes determines the maximum ID value. '*#*' represents the number of IDs loaded from the file and *addedIDs* is the number of new IDs (if appending to an existing list)
Warning	At least one subfolder must be selected	None of the targeted source code folder's subfolders are selected to be searched in the Search Source dialog
Warning	Cannot close ID file *path+filename*	An I/O exception occurred while attempting to close the ID file
Warning	Cannot close log file *path+filename*	An I/O exception occurred while attempting to close the log file
Error	Cannot create ID file *path+filename*	An I/O exception occurred while attempting to create the ID file, which can be due to the target folder not having write permission enabled
Error	Cannot create temporary file or log notes file *path+filename*	An error occurred while creating the temporary file or the log notes file, which can be due to the target folders not having write permission enabled
Error	Cannot delete original log notes file *path+filename*	An error occurred while attempting to delete the original log notes file (in preparation for replacing it with the new file), which can be due to the notes file folder not having write permission enabled
Error	Cannot delete temporary file for log notes file *path+filename*	An error occurred while attempting to delete the empty temporary log notes file, which can be due to the temporary folder not having write permission enabled
Error	Cannot locate ID file *path+filename*	The performance ID file chosen by the user in the Load Performance IDs dialog cannot be found in the selected folder

Type	Message	Cause
Error	Cannot locate log file *path+filename*	The performance log file chosen by the user in the Read Log dialog cannot be found in the selected folder
Error	Cannot locate temporary file or log notes file *path+filename*	An "file not found" exception occurred while attempting to write the temporary file or read the log notes file, which can be due to the target folder not having read permission enabled
Error	Cannot read ID file *path+filename*	An I/O exception occurred while attempting to read the ID file, which can be due to the file not having read permission enabled
Error	Cannot read log file *path+filename*	An I/O exception occurred while attempting to read the log file, which can be due to the file not having read permission enabled
Error	Cannot read log notes file *path+filename*	An I/O exception occurred while attempting to read the log notes file, which can be due to the file not having read permission enabled
Error	Cannot rename temporary file to log notes file *path+filename*	An error occurred while renaming the temporary file to the log notes file
Error	Cannot replace ID file *path+filename*	The existing ID file cannot be replaced, which can be due to the target folder not having write permission enabled
Error	Cannot write temporary file or read log notes file *path+filename*	The log notes file can't be read or its temporary file can't be written, which can be due to the files' folders not having read or write permission enabled, respectively
Error	Config name cannot be blank	The Config name field in the Save Config dialog is empty or contains only whitespace characters (spaces)
Error	Content type mismatch for log file *path+filename*	The content type read from the log file header does not match the expected pattern ("cFE1"), which can be due file corruption or if the file is not a performance log. If multiple logs are being loaded this log is skipped, but the others are read
Error	Duplicate ID "*ID*" – matches row *rowNumber* [row #]	The performance ID entered in the ID column of the ID editor is the same as one already in the table. '#' indicates the row in the table where the error exists, and *rowNumber* indicates the row containing the duplicate ID
Error	Duplicate name "*IDname*" – matches row *rowNumber* [row #]	The performance ID name entered in the Name column of the ID editor is the same as one already in the table. '#' indicates the row in the table where the error exists, and *rowNumber* indicates the row containing the duplicate ID name

Type	Message	Cause
Warning	Error accessing program preferences	The program preference keys could not be read from the preference storage node
Error	Error reading description for log file *path+filename*	The description could not be read from the log file header, which can be due file corruption or if the file is not a performance log. If multiple logs are being loaded this log is skipped, but the others are read
Error	Error reading header data for log file *path+filename*	The data parameters could not be read from the log file header, which can be due file corruption or if the file is not a performance log. If multiple logs are being loaded this log is skipped, but the others are read
Error	Error reading masks for log file *path+filename*	The filter and/or trigger mask parameters could not be read from the log file header, which can be due file corruption or if the file is not a performance log. If multiple logs are being loaded this log is skipped, but the others are read
Error	Error reading metadata for log file *path+filename*	The metadata parameters could not be read from the log file header, which can be due file corruption or if the file is not a performance log. If multiple logs are being loaded this log is skipped, but the others are read
Error	Error searching for IDs	An error occurred while searching the CFS source code for performance IDs. This can be due to the targeted folder not containing the CFS source code, no IDs being defined in the code, or to an I/O error occurring while reading the header files
Error	Error writing temporary file or reading log notes file *path+filename*	An I/O exception occurred while attempting to write the temporary file or read the log notes file, which can be due to the target folders not having write or read permission enabled, or incorrect format of the original file's contents
Error	Error writing to ID file *path+filename*	An I/O exception occurred while attempting to write the ID file, which can be due to the target folder not having write permission enabled
Command Line Error	Error: *<command>* high bit cannot be set	The `minID` or `majID` command parameter's high bit (0x80000000) is set
Command Line Error	Error: *<command>* must be a color name or in the format '0x######' where '#' is a hexadecimal digit	The `backColor`, `majGridColor`, `minGridColor`, `minFrameColor`, or `majFrameColor` command parameter is not a valid color name or hexadecimal value, or does not contain 6 digits

Type	Message	Cause
Command Line Error	Error: <*command*> must be in the format '0x########' where '#' is a hexadecimal digit	The `minID` or `majID` command parameter contains non-hexadecimal digits or exceeds 8 digits in length
Command Line Error	Error: <*command*> must be 'true' or 'false'	The `labels`, `shading`, `hGrid`, `vGrid`, `gaps`, `overruns`, `idle`, `minFrame`, or `majFrame` command parameter is neither `true` or `false`
Command Line Error	Error: <*command*> must be >= # and <= #	The `height` or `spacing` command parameter is outside the allowable limits
Command Line Error	Error: <*command*> must be 'step', 'thread', or 'statistics'	The `plot` command parameter is not `step`, `thread`, or `statistics`
Command Line Error	Error: <*command*> not a number	The `height` or `spacing` command parameter contains a non-numeric (0-9) character
Command Line Error	Error: mainsize width or height not a number, or too many/few values	The width or height contains a non-numeric (0-9) character, or other than 2 values are given
Command Line Error	Error: order must be 'name' or 'value'	The `order` command parameter is neither `name` or `value`
Error	Frequency invalid (blank, negative, or not a number) [row #]	The performance ID frequency entered in the Freq column of the ID editor is empty, is a negative number, or contains non-numeric characters. '#' indicates the row in the table where the error exists
Error	Help file filename cannot be opened; Desktop class unsupported	The CPM help documentation file cannot be opened. This is due to the Java Desktop class not being available in the operating system
Error	Help file filename cannot be opened; file missing	The CPM help documentation file cannot be opened. This is due to the file not being included in the CPM.jar file
Error	Help file filename cannot be opened; no application registered to open .pdf files	The CPM help documentation file cannot be opened. This is due to having no application registered in the operating system to open .pdf files (the help file is in PDF format)
Error	I/O error obtaining the header file list	An error occurred while searching the CFS source code for performance IDs when an attempt was made to get the list of header files within a subfolder
Error	ID "*ID*" invalid (not a number) [row #]	The performance ID entered in the ID column of the ID editor is not a valid number (base 10 or base 16; note that base 16 numbers must be preceded by 0x). '#' indicates the row in the table where the error exists

Type	Message	Cause
Warning	ID "*ID*" value exceeds the maximum supported (*maxID*) [row #]	This message can only occur if a log is loaded. The performance ID value entered in the ID column of the ID editor exceeds the maximum ID value supported by the current log. The maximum ID value supported, *maxID*, is equal to the filter/trigger mask size (reported in the log header) multiplied by 32, minus 1. If multiple logs are loaded which have different mask sizes then the largest of the mask sizes determines the maximum ID value. '*#*' indicates the row in the table where the error exists
Error	ID must be provided for "*name*" [row #]	A row in the ID editor has an ID name (*name*) in the **Name** column but no ID in the **ID** column. '*#*' indicates the row in the table where the error exists
Warning	Log inconsistencies detected: - # instance(s) of a time stamp occurring out of order - # instance(s) of an ID with consecutive entry or exit events	'*#*' indicates the number of violations. Condition is possibly due to log corruption or use of incorrect CFS constants. Consecutive entry/exit events may be due to the placement of the performance ID entry and exit commands within the CFS code
Error	Minimum x-value must be less than the maximum x-value	The x-axis minimum value is equal to or greater than the maximum value in the Set Bounds dialog
Error	Out of memory error for log file *path+filename*	An out of memory error occurred while attempting to read the log file header, which can be due file corruption or if the file is not a performance log. If multiple logs are being loaded this log is skipped, but the others are read
Warning	Plot save failed	The attempt to save the plot, via the context menu, in PNG format failed. This may be due to the target folder (or file, if overwriting) not having write permission enabled
Warning	Problem occurred when setting the look & feel to *look&feel*	An exception occurred while attempting to set the look & feel to the one selected. This can occur if the look & feel is not supported by the platform, or if there is a problem with access to the look & feel information
Warning	Table *table* printing failed	Output of the table to a printer or file was unsuccessful. This can be due to the printer being offline
Error	X-axis values cannot be blank or contain non-numeric characters	The x-axis minimum and/or maximum value in the Set Bounds dialog contains non-numeric characters, is blank, or contains only whitespace (e.g., spaces)

Appendix D. Known Issues

1. If the user lacks administrator privileges then when the program starts in Windows a message similar to the following may be displayed at the command prompt:

   ```
   Sep 10, 2014 3:06:17 PM java.util.prefs.WindowsPreferences <init>

   WARNING: Could not open/create prefs root node Software\JavaSoft\Prefs
   at root 0x80000002. Windows RegCreateKeyEx(...) returned error code 5.
   ```

   This is a result of Windows attempting to create a global registry entry for the program preferences, even though only a user entry is requested. The user entry is successfully created/updated, so the warning message may be ignored. The message can be eliminated by executing the application once as an administrator since this adds the missing key. Adding the `Prefs` key manually is also an option.

2. When using the GTK+ look and feel in Linux, or any look and feel in Windows, the Files selection box does not highlight the files initially selected when the file choosing dialog is opened. The file name list does reflect the currently selected files, however.

3. When using certain Microsoft wireless mice running under Microsoft Windows the mouse wheel rotation is misinterpreted in Java applications. The effect in CPM is that when the mouse wheel is used to zoom in on a plot it causes the plot to zoom out instead. The issue has to do with the higher resolution capabilities of these mice. To allow a mouse with this problem to work properly with Java perform the following steps (note that if the scrolling problem returns following a reboot, then uninstall the mouse and mouse drivers and redetect the mouse - in Device Manager the mouse description should show as "HID-compliant mouse"; the steps below can then be performed):
   a. **Control Panel → Mouse**
   b. **Mouse Properties → Hardware** tab
   c. Select the problematic mouse from the list ("HID-compliant mouse")
   d. Click the **Properties** button
   e. Go to the **Details** tab
   f. Select "Device Instance Path" from the combo box
   g. A value will be displayed (e.g.: HID\VID_045E&PID_0745&MI_01&COL01\8&5538EC&0&0000); note this value. This is the path of the registry key that corresponds to this instance of the mouse
   h. Open the registry editor and navigate to:
   a. HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum*<value noted in step 7>*\Device Parameters
   i. In Device Parameters, add the following DWORD (32 bit) registry keys:
   a. HScrollHighResolutionDisable = 1
   b. VScrollHighResolutionDisable = 1
   c. Delta = 120 (decimal)
   j. Unplug, then plug back in the mouse transceiver to re-initialize the driver
   k. The wheel scrolling should work in Java after this. If the scroll speed is too fast then perform the remaining steps
   l. **Control Panel → Mouse**
   m. **Mouse Properties → Wheel**

n. Under **Vertical Scrolling** set "Roll the wheel one notch to scroll: The following number of lines at a time:" to 1

o. Select the **OK** button

p. Open the **Mouse and Keyboard Center**

q. Under **Basic Settings** select **Wheel**

r. Adjust the **Wheel Vertical Scrolling** slider to the slowest setting

Appendix E.   Program Notes

a.   CPM class files

Following is a list and description of the Java class files specific to the CPM application.

CPMAppearanceDialog.java	Class that creates and manages the Appearance dialog used for selecting the application look & feel. The dialog is built on the CPMDialogHandler class
CPMCommandLineHandler.java	Class for reading and executing the command line options
CPMConfigDialog.java	Class that creates and manages the Load Config, Save Config, and Delete Config dialogs. The dialog is built on the CPMDialogHandler class
CPMConstants.java	Class containing constant values used by the other classes
CPMDialogHandler.java	Generic utility class for creating and handling all of the dialogs created within the application. Dialog box types handled include status messages, color choice, and ones with user-created controls and/or buttons
CPMIDEditorDialog.java	Class that creates and manages the ID Editor dialog. The dialog is built on the CPMDialogHandler class
CPMIDHandler.java	Class that handles all of the ID menu items. Includes methods for manipulation of the performance IDs, such as searching the CFS source code for IDs
CPMLogContentDialog.java	Class that creates and manages the Log Data dialog. The dialog is built on the CPMDialogHandler class
CPMLogHandler.java	Class that handles all of the Log menu items. Includes methods for manipulation of the performance log data, such as statistics calculations
CPMLogHeaderDialog.java	Class that creates and manages the Log Header dialog. The dialog is built on the CPMDialogHandler class
CPMLogStatisticsDialog.java	Class that creates and manages the Log Statistics dialog. The dialog is built on the CPMDialogHandler class
CPMMain.java	The CPM main application class handles flow and execution of the menu bar items and plot tabs
CPMPreferencesDialog.java	Class that creates and manages the Plot Preferences dialog. The dialog is built on the CPMDialogHandler class
CPMPreferencesHandler.java	Class for managing the plot preferences including setting the default values, and building and parsing the preferences information stored in the program preferences location

CPMSetBoundsDialog.java   Class that creates and manages the Set Bounds dialog used to adjust the step and thread of execution x- and y-axis boundaries. The dialog is built on the CPMDialogHandler class

CPMSourceSearchDialog.java   Class that creates and manages the source code search criteria dialog. The dialog is built on the CPMDialogHandler class

CPMStatisticsPlot.java   Class that creates and manages the statistics pie and bar charts

CPMStepPlot.java   Class that creates and manages the step plot. The plot is built on the CPMXYPlotHandler class

CPMTableHandler.java   Generic utility class for creating and handling all of the tables created within the application, including the log data, log header. Log statistics, and ID editor tables

CPMThreadPlot.java   Class that creates and manages the thread of execution plot. The plot is built on the CPMXYPlotHandler class

CPMXYLineAndShapeRenderer.java   Contains methods from the JFreeChart library overridden to skip plotting of points coincident on the display in order to improve plot draw performance

CPMXYPlotHandler.java   Generic utility class for creating and handling the step and thread of execution X-Y plots

CPMXYStepRenderer.java   Contains methods from the JFreeChart library overridden to produce the shading beneath the step plot lines
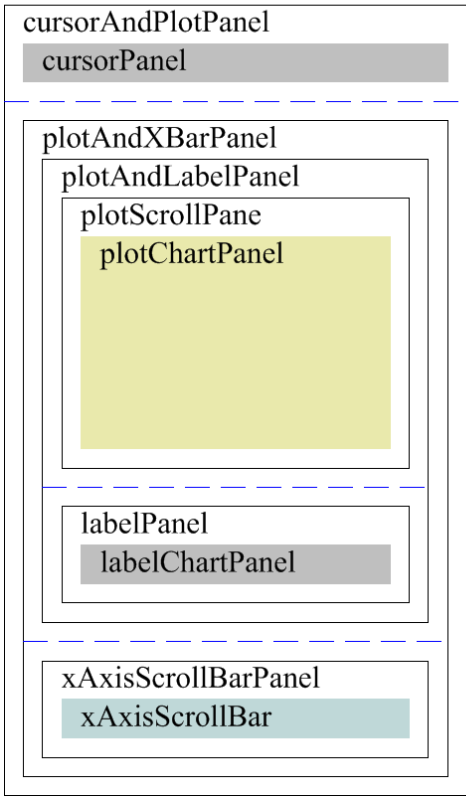
b. Use of long integers

As of Java 1.7 unsigned variable types are not directly supported. Normally unsigned integers would be used for the log header time stamp values and performance IDs, but this leads to sign-related issues for larger values. Instead, long integers are used, which avoids these sign-related issues.

c. X-Y plot component layout

Figure 34 shows the relationship of the Java Swing components used to arrange and display the step and thread of execution plots. The component labels are the variable names used in the application source code. The use of borders allows the layout manager to (re)position and (re)size the components as the main application window is resized. For example, the position of the x-axis scroll bar is dependent on the length of the performance ID names used for the y-axis labels. Certain components are contained within a JPanel container in order for the selected L&F to respond properly to main window resizing. The difference between the step and thread of execution plots, both of which are based on the CPMXYPlotHandler class, is the y-axis scroll bar. The step plot relies on the y-axis scroll bar provided by the scroll pane in which the plot resides, whereas the thread of execution plot requires the use of custom y-axis scroll bar.

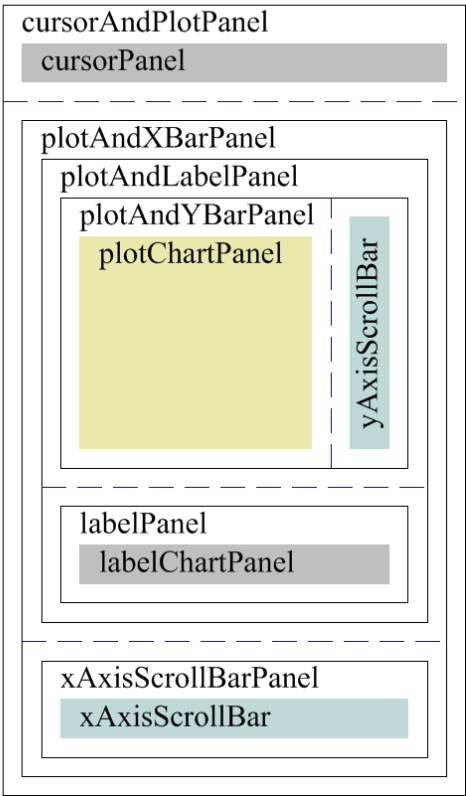### Step Plot Component Layout   Thread Plot Component Layout

Figure 34. Nesting of step and thread of execution plot Java Swing components