NightTune RT User's Guide

Version 2.2

(RedHawk™ Linux®)

concurrent

0898515-020
July 2006

The information contained in this document is believed to be correct at the time of publication. It is subject to change without notice. Concurrent Computer Corporation makes no warranties, expressed or implied, concerning the information contained in this document.

To report an error or comment on a specific portion of the manual, photocopy the page in question and mark the correction or comment on the copy. Mail the copy (and any additional comments) to Concurrent Computer Corporation, 2881 Gateway Drive, Pompano Beach, FL 33069-4324. Mark the envelope **"Attention: Publications Department**." This publication may not be reproduced for any other reason in any form without written permission of the publisher.

Preface

Scope of Manual

This guide is designed to assist you in getting started with use of NightTune™, a process and system analysis and tuning tool.

Structure of Manual

This manual consists of four chapters and an index. A brief description of the contents of each of the parts of the manual follows.

- Chapter 1 introduces NightTune, its command line options, and system requirements.
- Chapter 2 describes NightTune display windows.
- Chapter 3 describes each of NightTune's functional display panels.
- Chapter 4 describes how to operate NightTune to execute specific monitoring and tuning tasks.

Syntax Notation

The following notation is used throughout this guide:

italic	Books, reference cards, and items that the user must specify appear in *italic* type. Special terms and comments in code may also appear in *italic*.
`list bold`	User input appears in **`list bold`** type and must be entered exactly as shown. Names of directories, files, commands, options and man page references also appear in **`list bold`** type.
`list`	Operating system and program output such as prompts and messages and listings of files and programs appears in `list` type. Keywords also appear in `list` type.
window	Keyboard sequences and window features such as push buttons, radio buttons, menu items, labels, and titles appear in window type.
`[ ]`	Brackets enclose command options and arguments that are optional. You do not type the brackets if you choose to specify such option or arguments.
`{ }`	Braces enclose mutually exclusive choices separated by the pipe (|) character, where one choice must be selected. You do not type the braces or the pipe character with the choice.
`...`	An ellipsis follows an item that can be repeated.

Referenced Publications

The following publications are referenced in this document:

0890514	*NightBench™ User's Guide*
0898004	*RedHawk Linux User's Guide*
0898008	*NightStar RT Installation Guide*
0898008	*NightStar RT Tutorial*
0898395	*NightView™ RT User's Guide*
0898398	*NightTrace™ RT User's Guide*
0898458	*NightSim™ RT User's Guide*
0898465	*NightProbe™ RT User's Guide*

Contents

Index

Illustrations

1
Introducing NightTune

NightTune's graphical user interface (GUI) provides a powerful and intuitive point-and-click style of operation that allows you to analyze and adjust system activities with ease.

NightTune has the following features:

- Process monitoring and tuning

  Through the Process List and Process Monitor panels you can select individual processes from a list and monitor their CPU time, memory size, scheduling parameters, CPU affinity and other process attributes. You can monitor the same attributes of individual threads within a process. You can modify the scheduling parameters and CPU affinity with the pop-up Process Scheduler dialog or with drag-and-drop operations to the CPU Status panel.

- CPU control

  You can monitor the status of a CPU, including information about shielding and process and interrupt bindings. If you have the appropriate privileges, you can set shielding and hyper-threading attributes for each CPU.

- System monitoring and tuning

  With the system activity panels, you can monitor:

  - Processor Usage
  - Context Switches
  - Virtual Memory
  - Disk Activity
  - Network Activity
  - Interrupt Activity

  Information is available both numerically and graphically. The Interrupt Activity panel allows you to change the CPU affinity for individual interrupts.

- Tuning Activity log

  This facility provides optional logging of tuning activity to a user-specified log file.

- Comprehensive online help

  The help system includes context-sensitive help, accessible by clicking on any item in NightTune's window.

NightTune uses the NightStar License Manager (NSLM) to control access to the NightStar RTtools. See "NightStar Licensing" on page A-1 for more information.

Point and Click Operation

NightTune provides a point-and-click interface. Most operations utilize the mouse.

NightTune makes extensive use of right-click pop-up menus and drag-and-drop actions.

Local or Remote Operation

NightTune can operate on local or remote systems. When operating remotely, the graphical user interface runs on the host system where NightTune is invoked and communicates to a NightTune server process which is launched automatically on the remote target system.

The Pluggable Authentication Module (PAM) is used to authenticate connection requests to remote systems (see **`pam(8)`**). The file **`/etc/pam.d/ntune`** is installed as part of the NightTune product; it defines the specific authentication mechanisms that are used for each target system. For security reasons, NightTune encrypts usernames and passwords during all authentication requests and never stores authentication information to disk.

See "Command Line Options" on page 1-4 for more information on remote operation of NightTune.

Capabilities

Most operations with NightTune do not require any special privileges. However, if you wish to modify the scheduling attributes of other user's processes, or adjust CPU shielding attributes, you must have the `CAP_SYS_NICE` capability or invoke NightTune as the `root` user.

Linux provides a means to grant otherwise unprivileged users the authority to perform certain privileged operations. The Pluggable Authentication Module (see **`pam_capability(8)`**) is used to manage sets of capabilities, called *roles*, required for various activities.

Linux systems should be configured with an `ntuneuser` role which provides the `CAP_SYS_NICE` capability.

Edit **`/etc/security/capability.conf`** and define the `ntuneuser` role (if it is not already defined) in the "ROLES" section:

```
role ntuneuser CAP_SYS_NICE
```

Additionally, for each NightTune user on the target system, add the following line at the end of the file:

```
user   username     ntuneuser
```

where *username* is the login name of the user.

If the user requires capabilities not defined in the `ntuneuser` role, add a new role which contains `ntuneuser` and the additional capabilities needed, and substitute the new role name for `ntuneuser` in the text above.

In addition to registering your login name in **`/etc/security/capability.conf`**, certain files under the **`/etc/pam.d`** directory must also be configured to allow capabilities to be activated.

To activate capabilities, add the following line to the end of selected files in **`/etc/pam.d`** if it is not already present:

```
session required pam_capability.so
```

The list of files to modify is dependent on the list of methods that will be used to access the system. The following table presents a recommended configuration that will grant capabilities to users of the services most commonly employed in accessing a system.

Table 1-1. Recommended /etc/pam.d Configuration

/etc/pam.d File	Affected Services	Comment
`remote`	telnet rlogin rsh (when used w/o a command)	Depending on your system, the **`remote`** file may not exist. Do not create the **`remote`** file, but edit it if it is present.
`login`	local login (e.g. console) telnet* rlogin* rsh* (when used w/o a command)	*On some versions of Linux, the presence of the **`remote`** file limits the scope of the **`login`** file to local logins. In such cases, the other services listed here with **`login`** are then affected solely by the **`remote`** configuration file.
`rsh`	rsh (when used with a command)	e.g. **`rsh system_name a.out`**
`sshd`	ssh	You must also edit **`/etc/ssh/sshd_config`** and ensure that the following line is present: **`UsePrivilegeSeparation no`**
`gdm`	gnome sessions	
`kde`	kde sessions	

If you modify **`/etc/pam.d/sshd`** or **`/etc/ssh/sshd_config`**, you must resetart the **`sshd`** service for the changes to take effect:

```
service sshd restart
```

In order for the above changes to take effect, the user must log off and log back onto the target system.

NOTE

To verify that you have been granted capabilities, issue the following command:

```
/usr/sbin/getpcaps $$
```

The output from that command will list the roles currently assigned to you.

Command Line Options

Use the following command to start NightTune:

ntune [**-help**] [**-version**]
[**-config** *config-file*] [**-display** *display-name*]
[**-logfile** *logfile*] [**-target** *system-name*]

Options are described as follows:

-help

Causes NightTune to display its command line syntax (followed by a brief description of each option), then exit.

-version

Causes NightTune to display its current version, then exit.

-config *config-file*

Specifies the configuration file that contains the start-up information for NightTune. In the absence of the **-config** option, NightTune searches for a configuration file using the following ordered criteria:

- a file as specified in the `*mxtOptConfigFile` resource
- a **.NightTunerc** file in the current working directory
- a **.NightTunerc** file in the user's home directory

-display *display-name*

The name of your X11 graphical display terminal. This is typically a string such as "*server*`:`*n*" where "*server*" is the network address of your server, and "*n*" is the display number, typically 0.

If the **-display** option is omitted, the graphical display terminal will be selected based on the value of the `DISPLAY` environment variable.

You should not use other standard X11 command line options (e.g., **`-geometry`**).

`-logfile` *logfile*

Specifies the file to be used for logging tuning activity. If logging is enabled via the Logging option from the NightTune menu, tuning events are logged to this file. The name of the file can be entered on the command line or specified in the configuration file; a log file named on the command line overrides one specified in the configuration file.

`-target` *system-name*

Causes NightTune to operate on the specified remote target system. The NightTune GUI will run on the system where **`ntune`** was invoked and will communicate with a NightTune server process, which is automatically launched on the target system.

NightTune connects to the target system using the **`nstar.d`** daemon, supplied as part of NightStar RT tools, running on that target system.

2
NightTune Windows

NightTune features are available through a number of functional panels that are placed in tabbed pages in NightTune windows.

Much of the time you will operate NightTune with a single window displaying several panels of information within one or more tabbed pages. However, NightTune allows you to have multiple windows active at once, customized to display any panels of interest.

Each window has:

- a menu bar
- a display and control area, which is the main body of the window where tabbed pages display selected panels (see “NightTune Panels” on page 3-1 and “Drop Targets” on page 2-12)
- control buttons that affect how data is refreshed (see “Control Buttons” on page 2-13)

In addition, the Preferences dialog allows the user to select logging attributes, configure the CPU Status panel, and control the update intervals used by NightTune to refresh displayed data. See “Preferences” on page 2-14 for more information.

Menu Bar

The menu bar provides access to preference settings, launching additional tools, obtaining help, and controlling which panels are displayed in the window as well as the appearance of items inside panels.

The menu bar provides the following menus:

- NightTune
- Windows
- Monitor
- Frames
- Tools
- Help

Each menu is described in the sections that follow.

NightTune

Mnemonic: Alt+N

The NightTune menu allows you to set preferences, load or save configuration data, activate or deactivate logging, and contains the means to exit NightTune.

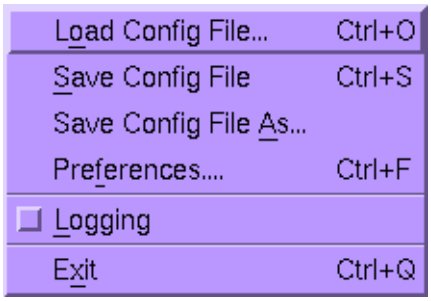

Figure 2-1. NightTune menu

The following paragraphs describe the options on the NightTune menu in more detail.

Load Config File...

Mnemonic: O
Accelerator: Ctrl+O

This menu item displays a file selection dialog. By selecting a previously-saved configuration file, NightTune closes all current windows and opens windows as defined in the configuration file.

Save Config File

Mnemonic: S
Accelerator: Ctrl+S

This menu item allows you to save the configuration data from the current session to a configuration file. Configuration data includes the layout of panels in tabbed pages for windows in the current session as well as settings in the Preferences dialog (see "Preferences" on page 2-14).

When invoked without the **`-config`** option, NightTune automatically searches for a configuration file. It first looks for **`.NightTunerc`** in the current working directory and then in your home directory. The first file found is automatically loaded when NightTune launches.

The name of the configuration file that will be saved will be the configuration file last saved or loaded. If no configuration file was ever associated with the current NightTune session, then the configuration file name will be **`.NightTunerc`** and will be placed in your home directory.

Save Config File As...

Mnemonic: A

This menu item allows you to save the configuration data from the current session to a configuration file of your choice.

When you select this menu item, NightTune displays a file selection dialog. After making a selection, the configuration data from the current session is saved in the selected file. Configuration data includes the layout of panels in tabbed pages for windows in the current session as well as settings in the Preferences dialog (see “Preferences” on page 2-14).

Preferences...

Mnemonic: F
Accelerator: Ctrl+F

This menu item launches the Preferences dialog which allows you to customize logging attributes and refresh intervals for various panels.

See “Preferences” on page 2-14 for more information.

Logging

Mnemonic: L

This menu item toggles the state of logging.

When logging is enabled, all modifications to processes scheduling attributes, CPU shielding settings, and interrupt CPU affinity are logged in ASCII text to a log file. The Process Monitor panel allows you to log detailed process information to the log file as well.

By default, the name of the log file is **`ntune.log`**, placed in the current working directory from which NightTune was launched. The name of the log file can be changed with the Preferences dialog.

Exit

Mnemonic: X
Accelerator: Ctrl+Q

This menu item exits NightTune. NightTune will not prompt you to save any unsaved configuration data before exiting -- select Save Config File prior to exiting to retain configuration data for subsequent invocations of NightTune.

Windows

Mnemonic: Alt+W

The Windows menu allows you to control tabbed pages and the number and iconified state of windows in the current NightTune session.

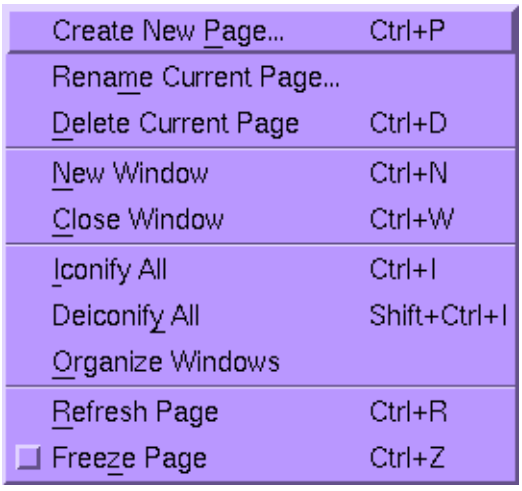

Figure 2-2. Windows menu

The following paragraphs describe the options on the Windows menu in more detail.

Create New Page...

Mnemonic: P
Accelerator: Ctrl+P

This menu item creates a new tabbed page in the current window. A dialog is launched to allow you to associate a name with the page.

Rename Current Page...

Mnemonic: M

This menu item opens a dialog which allows you to change the name associated with the tabbed page currently displayed in the window.

Delete Current Page

Mnemonic: D
Accelerator: Ctrl+D

This menu item deletes the tabbed page currently displayed in the window.

New Window

Mnemonic: N
Accelerator: Ctrl+N

This menu item opens a new window in the current NightTune session with a single tabbed page without any associated panels.

Close Window

Mnemonic: C
Accelerator: Ctrl+W

This menu item closes the current window. If the current window is the last NightTune window open, NightTune exits.

Iconify All

Mnemonic: I
Accelerator: Ctrl+I

This menu item iconifies all windows associated with the current NightTune session.

Deiconify All

Mnemonic: Y
Accelerator: Shift+Ctrl+I

This menu item restores all windows associated with the current NightTune session.

Organize Windows

Mnemonic: O

This menu item deiconifies all NightTune windows and repositions them on the display with a cascading orientation.

Refresh Page

Mnemonic: R
Accelerator: Ctrl+R

The Refresh menu item immediately refreshes the displayed data in all panels in the current tabbed page.

Freeze Page

Mnemonic: Z
Accelerator: Ctrl+Z

This menu item toggles the Freeze setting for all panels in the current tabbed page. When frozen, data values are not refreshed automatically; however, you can manu-

ally refresh the data on an otherwise frozen page by using the Refresh Page menu item.

Monitor

Mnemonic: Alt+M

The Monitor menu allows you to select which panels are to be displayed in the tabbed page currently showing in the NightTune display area.

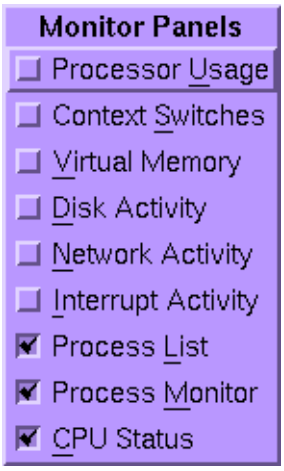

Figure 2-3. Monitor menu

NOTE

It is important to note that the Monitor menu, like the other menus, can be "torn off" from the menu bar to reside in its own window separate from NightTune. This is especially useful when making multiple changes to the panels so as to avoid selecting the menu after every choice.

Each of the menu items act as toggles which control whether the selected panel is visible in the tabbed page currently displayed in the NightTune display area.

The following paragraphs summarize individual panel activities. Detailed descriptions for each panel are provided in "NightTune Panels" on page 3-1.

Processor Usage

Mnemonic: U

This panel displays CPU utilization using textual and graphical displays. It provides User, System, Wait, and Idle time information per CPU.

Context Switches

Mnemonic: S

This panel displays context switches per CPU per second using textual and graphical displays.

Virtual Memory

Mnemonic: V

This panel displays system paging rates using textual and graphical displays.

Disk Activity

Mnemonic: D

This panel provides detailed disk activity for all disks in the system. It includes information about read and write operations and service times using textual and graphical displays.

Network Activity

Mnemonic: N

This panel describes network activity in terms of packet I/O rates, errors, and collisions for each network device on the system using textual and graphical displays.

Interrupt Activity

Mnemonic: I

This panel displays interrupt information for the system using textual and graphical displays. It allows you to change the CPU affinity of individual interrupts.

Process List

Mnemonic: L

This panel lists all processes on the system and organizes the processes by user.

Process Monitor

Mnemonic: M

This panel provides detailed process information for selected processes. It also allows you to change scheduling attributes for processes.

CPU Status

Mnemonic: C

This panel describes each CPU in the system and includes information on the state of shielding, process binding, and interrupt binding. It allows you to change the shielding attributes of CPUs and to redefine the CPU binding of specific processes or interrupts.

Frames

Mnemonic: Alt+F

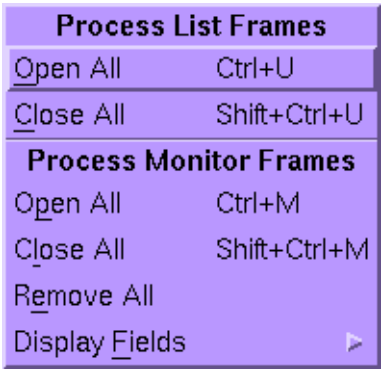

Figure 2-4. Frames menu

The Frames menu only affects the Process List or Process Monitor panels. These panels organize the information they present in "frames". The menu items control the frames.

The following describe the options on the Frames menu:

Open All

Mnemonics: O, P
Accelerators: Ctrl+U, Ctrl+M

Expands the contents of all frames on the associated panel of the current tabbed page.

Close All

Mnemonics: C, L
Accelerators: Shift+Ctrl+U, Shift+Ctrl+M

Collapses the expanse of individual information for all frames on the associated panel of the current tabbed page. The frames are not removed from the panel.

Remove All

Mnemonic: E

Removes all frames from the Process Monitor panel on the current tabbed page. They can be added to the panel again using the Process List panel.

Display Fields

Mnemonic: F

Opens the Process Fields menu which allows you to select the process attributes that you want to be displayed for processes in frames on the Process Monitor panel. The Process Fields menu will be described in detail in the chapter describing the Process Monitor panel.

Customization of Process Fields affects all current and future Process Monitor panels in all tabbed pages of all windows.

Tools

Mnemonic: Alt+T

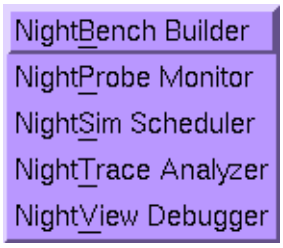

Figure 2-5. Tools menu

The following describe the options on the Tools menu:

NightBench Builder

Mnemonic: B

Opens the NightBench Program Development Environment. NightBench is a set of graphical user interface (GUI) tools for developing software with the Concurrent MAXAda™ compiler toolset.

See also:

- *NightBench User's Guide*

NightProbe Monitor

Mnemonic: P

Opens the NightProbe Data Monitoring tool. NightProbe is a real-time graphical tool for monitoring, recording, and altering program data within one or more execut-

ing programs without significant intrusion. NightProbe can be used in a development environment as a tool for debugging or in a production environment for data capture or to create a "control panel" for program input and output.

See also:

- *NightProbe RT User's Guide*

NightSim Scheduler

Mnemonic: S

Opens the NightSim Application Scheduler. NightSim is a tool for scheduling and monitoring real-time applications which require predictable, repetitive process execution. With NightSim, application builders can control and dynamically adjust the periodic execution of multiple coordinated processes, their priorities, and their CPU assignments.

See also:

- *NightSim RT User's Guide*

NightTrace Analyzer

Mnemonic: T

Opens the NightTrace Analyzer. The NightTrace Analyzer is a graphical tool for analyzing the dynamic behavior of multi-process and/or multi-processor user applications and operating system activity. NightTrace allows you to control user and kernel trace collection daemons and can graphically display the interplay between many real-time programs and processes across multiple processors and systems.

See also:

- *NightTrace RT User's Guide*

NightView Debugger

Mnemonic: V

Opens the NightView Source-Level Debugger. NightView is a graphical source-level debugging and monitoring tool specifically designed for real-time applications. NightView can monitor, debug, and patch multiple real-time processes running on multiple processors with minimal intrusion.

See also:

- *NightView RT User's Guide*

Help

Mnemonic: Alt+H

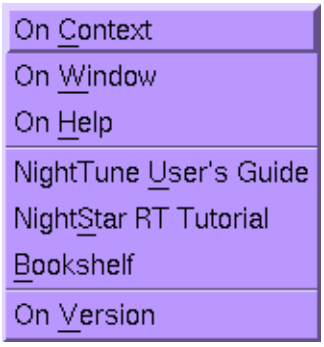

Figure 2-6. Help menu

NightTune uses HyperHelp™, an integrated online help system which provides hyperlinks and context-sensitive help.

Using HyperHelp is intuitive and easy.

The following describe the options on the Help menu:

On Context

Mnemonic: C

Gives context-sensitive help on the various menu items, dialogs, or other parts of the user interface.

Help for a particular item is obtained by first choosing this menu item, then clicking the mouse pointer on the object for which help is desired. The mouse pointer becomes a floating question mark when the On Context menu item is selected.

In addition, context-sensitive help may be obtained for the widget with the current focus by pressing the F1 key. HyperHelp, NightTune's online help system, will open with the appropriate topic displayed.

On Window

Mnemonic: W

Displays help information for the current window.

On Help

Mnemonic: H

Displays this section of the on-line manual.

NightTune User's Guide

Mnemonic: U

Opens the online version of the *NightTune User's Guide* in the HyperHelp viewer.

NightStar RT Tutorial

Mnemonic: S

Opens the online version of the *NightStar RT Tutorial* in the HyperHelp viewer.

Bookshelf

Mnemonic: B

Opens a HyperHelp window that lists all of the currently available HyperHelp publications.

On Version

Mnemonic: V

Displays a short description of the current version of NightTune.

Drop Targets

NightTune windows have two drop targets for use with the various panels.

Kill

If a process or thread is dragged to this drop target, then that process or thread is killed with a `SIGKILL`.

Unbind

If a process, thread, or interrupt is dragged to this drop target, then that process, thread, or interrupt is unbound from any CPUs. That is, its CPU affinity is set to include all CPUs.

Note that interrupts can be dragged to this drop target only if NightTune is running as `root` or if the user running NightTune has been granted the proper capabilities (see "Capabilities" on page 1-2).

Control Buttons

NightTune windows have two control buttons near the bottom of the window that control how displayed data is refreshed within panels.

Freeze

Selecting the Freeze checkbox halts the automatic refresh of displayed data within the Window. It affects all panels within the window, however, individual panels can subsequently override the window frozen state using pop-up menus.

Refresh

Clicking the Refresh button causes all data displayed in all panels associated with the window to be refreshed once, regardless of the state of the Freeze checkbox on the window or in individual panels.

Preferences

The Preferences dialog allows you to tailor aspects of how NightTune operates.

These preferences are part of NightTune configuration data which can be saved to a configuration file so that subsequent invocations of NightTune can use the customized settings.

The following illustrates the Preferences dialog:

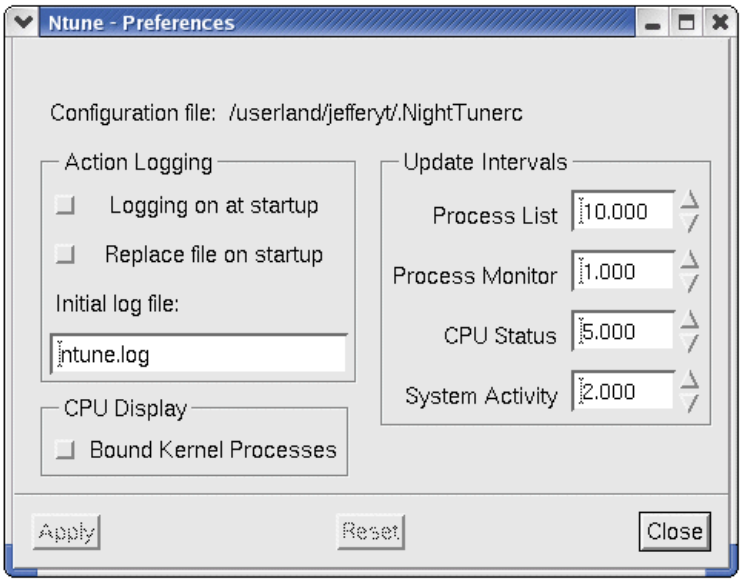

Figure 2-7. Preferences dialog

The following paragraphs describe the labels and control features of this dialog:

Configuration File

This area shows the name of the current configuration file. The file name can be changed using the Save Config File As... menu item from the NightTune menu.

Action Logging

This area allows you to select logging attributes.

Logging on at startup

This checkbox controls whether logging is automatically initiated when NightTune is launched. The logging status is controlled using the Logging option of the NightTune menu.

Replace file on startup

This checkbox controls whether the log file should be truncated when NightTune is launched. It has no effect unless Logging on at startup is activated.

Initial log file:

This text area allows you to specify the name of the log file. This setting is overridden by the **`-logfile`** command line option.

CPU Display

This area allows you to configure the CPU Status panel.

Bound Kernel Preferences

This checkbox controls whether or not the CPU Status panel shows kernel processes bound to particular CPUs. The RedHawk Linux kernel includes some processes which are visible to the user (e.g. **`ksoftirqd/0`**). Generally, users do not wish to see these processes, and so the default behavior is to exclude them from the CPU Status panel. But if this checkbox is checked, they will be displayed as appropriate for their CPU binding.

Update Intervals

This area allows you to control the update intervals used by NightTune to refresh displayed data. The units of all intervals are in seconds and can include fractional values.

An update interval of zero indicates that automatic updates will not occur. In such an instance, displays will only be updated when you click the Refresh button or select a Refresh option from panel pop-up menus.

Process List

The interval between which the Process List panel scans for new processes and updates user frames.

Process Monitor

The interval between which the Process Monitor process lines are updated.

CPU Status

The interval between which the shielding attributes and bound process and interrupt lists are updated in the CPU Status panel.

System Activity

The interval between which data within the Interrupt Activity, Context Switches, Processor Usage, Virtual Memory, Disk Activity, and Network Activity panels are updated.

3
NightTune Panels

This chapter describes NightTune's panels which provide functional units for displaying and modifying process and system activities.

The panels include:

- Process List (see "Process List Panel" on page 3-2)
- Process Monitor (see "Process Monitor Panel" on page 3-6)
- CPU Status (see "CPU Status Panel" on page 3-17)
- Interrupt Activity (see "Interrupt Activity Panel" on page 3-24)
- Processor Usage (see "Processor Usage Panel" on page 3-31)
- Context Switches (see "Context Switches Panel" on page 3-36)
- Virtual Memory Activity (see "Virtual Memory Activity Panel" on page 3-40)
- Disk Activity (see "Disk Activity Panel" on page 3-43)
- Network Activity (see "Network Activity Panel" on page 3-46)

Process List Panel

The purpose of the Process List panel is to provide a summary of active processes and to allow selection of processes for detailed monitoring in the Process Monitor panel.

The process list is organized by grouping processes by user ID within frames.

The following illustrates the Process List panel:

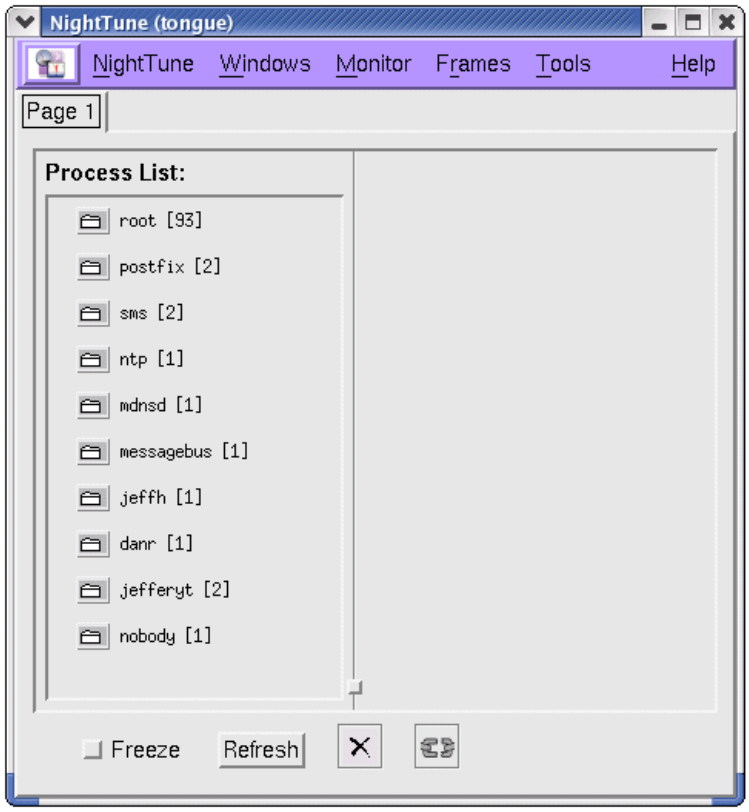

Figure 3-1. Process List panel

The panel displays a User Frame for each user with active processes.

User Frames

Each user frame contains the following information:

Folder Icon

A folder icon which represents a list of individual processes for the user. Clicking on the folder opens the list. The icon acts as a toggle; subsequent clicking closes and opens the folder, hiding or exposing the list of processes.

User Name

The name of the user is displayed.

Number of Processes

The number of processes inside each frame is displayed inside square brackets to the right of the user name.

The list is automatically refreshed at a selectable interval or by clicking the Refresh button at the bottom of the pane. The refresh interval can be changed using the Preferences dialog from the Preferences… item in the NightTune menu.

When the number of processes associated with a user frame drops to zero, the user frame is automatically deleted from the panel. As new user processes appear, new user frames are automatically added to the panel.

The following four items are displayed for each process in a user frame when the frame's folder is opened:

Checkbox

The checkbox controls whether the process is displayed in the Process Monitor panel.

PID

The process ID of the process is displayed.

Time

The combined system and user time associated with the process is displayed in units of seconds.

Process Name

The simple name of the process is displayed.

Process List Drag and Drop Operations

Individual processes or groups of processes can be dragged onto various destination panels and drop targets.

To drag an individual process, middle-click anywhere on the row describing the process and drag the pointer to the destination area and release the mouse button. Similarly, you can drag all the processes currently associated with a user by middle-clicking on the folder icon of a user frame and dragging the pointer to the destination area.

The Process List panel supports the following drag and drop operations:

- Dragging a process or user frame onto a CPU box in the CPU Status panel binds the processes to the corresponding CPU.
- Dragging a process or user frame to the Process Monitor panel causes the processes to be displayed in that panel.
- Dragging a process onto the Process Scheduler dialog of the Process Monitor panel changes the dialog to refer to that process.
- Dragging a process or user frame onto the Kill drop target causes the processes to be killed with a **SIGKILL** signal.
- Dragging a process or user frame onto the Unbind drop target causes the processes to be unbound from any CPUs.

When dragging processes or user frames, only the current processes associated with the operation are affected. If new processes subsequently are executed for the user, they will appear in the user frame in the Process List panel, but they are not automatically displayed in the Process Monitor panel or in the destination of the drop operation.

Process List Pop-up Menu

To launch the Process List pop-up menu, right-click while positioned in the Process List panel.

Figure 3-2. Process List pop-up menu

The following paragraphs describe the options on the menu in more detail:

Freeze/UnFreeze Display

This menu item toggles the Freeze setting for this panel. When frozen, this panel is not refreshed automatically.

Refresh Display

This menu item causes the panel to refresh with the latest list of users and processes, regardless of the Freeze setting.

Open all frames

This menu item opens all user frame folders, displaying all processes on the system.

Close all frames

This menu item closes all user frame folders, hiding individual processes.

Process Monitor Panel

The Process Monitor provides detailed descriptions of individual processes and threads.

The following illustrates the Process Monitor panel:

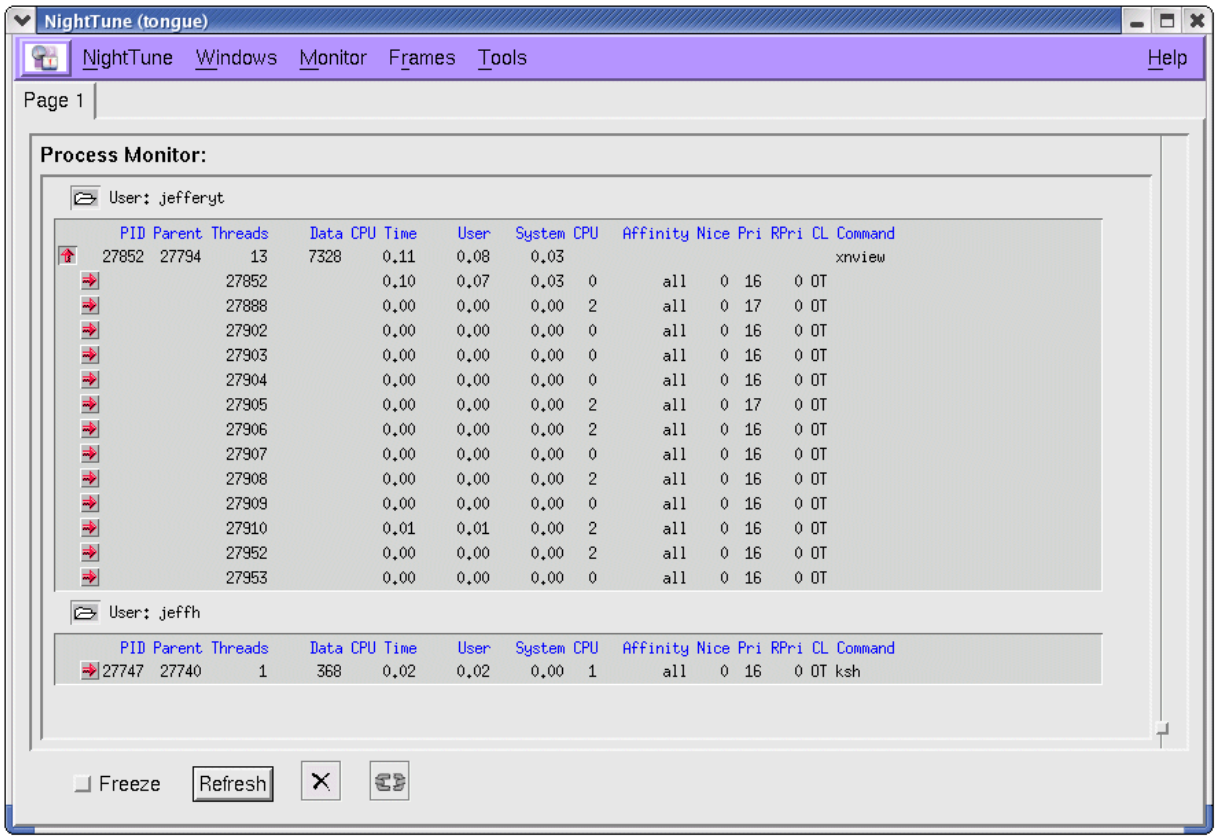

Figure 3-3. Process Monitor panel

This panel groups processes into the following types of process monitoring frames:

User Frames

A user frame holds individual processes for a single user. Processes are associated with this frame using the Process List panel. Processes are not automatically added to the frame when they begin execution. They must be selected in the Process List panel. However, when a process exits, it is automatically removed from the user frame.

CPU Frames

A CPU frame collects individual processes which are bound to a specific CPU. CPU frames are added to the panel from the CPU Status panel. Unlike user frames, the

list of processes in a CPU frame is dynamically updated as process bindings to the corresponding CPU change.

Clicking on the folder icon associated with each frame toggles the expansion of the list of processes associated with that frame.

Each frame contains the following items for each process being monitored:

Toggle Button (vertical arrow)

For processes with multiple threads, a toggle button with a vertical arrow controls the display of thread information for the process. This button appears only for processes with multiple threads. When you click an arrow button that points down, the process information is expanded to include an individual line for each thread in the process. When you click an arrow button that points up, the list of individual threads is hidden. For multi-threaded processes, the process line provides summary information for all threads within the process, regardless of the direction of the arrow button.

Push Button (right arrow)

The push button with the right red arrow activates the Process Scheduler dialog which allows you to change scheduling attributes of the process or thread.

Process Line

The process line shows detailed process information. The information displayed is controlled from the Process Fields menu from the Display Fields item of the Frames menu.

Information in the process line is automatically updated at a selectable interval. The interval can be changed using the Preferences dialog from the Preferences… item from the NightTune menu. Immediate refresh of information is available using the Refresh button at the bottom of the window or through the Process Monitor Panel pop-up menu.

Some operations in the panel operate on processes that are selected in the process list. Selection of individual processes is done using the left mouse button, which toggles whether the associated process is selected or deselected. Multiple selection is done by selecting individual processes, or using the Process Monitor pop-up menu to select all processes in the frame. Selecting or deselecting a multi-threaded process selects or deselects all its threads.

Process Fields Menu

The Process Fields menu is launched from the Display Fields option of the Frames menu. It applies to all current and future frames in all Process Monitor panels in all windows.

The following illustrates the Process Fields menu:

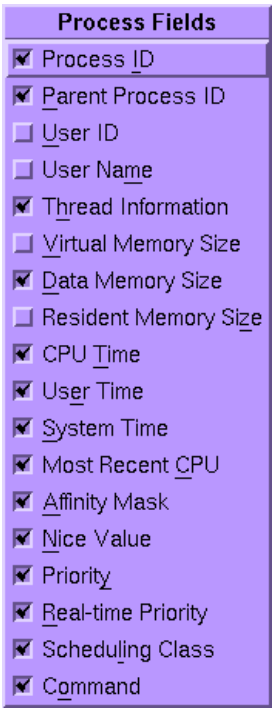

Figure 3-4. Process Fields menu

NOTE

It is important to note that the Process Fields menu, like the other menus, can be "torn off" from the menu bar to reside in its own window separate from NightTune. This is especially useful when making multiple changes so as to avoid selecting the menu after every choice.

The following paragraphs describe each of the selectable fields in more detail:

Process ID

This field displays the process ID as returned by **`getpid(2)`**.

Parent Process ID

This field displays the process ID of the parent process.

User ID

The field displays the user ID of the process

User Name

This field displays the user name corresponding to the user ID

Thread Information

For a process line, this field displays the number of threads in the process. For a thread line, this field displays the thread ID as returned by **`gettid(2)`**.

Virtual Memory Size

This field displays the amount of virtual memory associated with the process in units of kilobytes (KB)

Data Memory Size

This field represents an approximation of the memory used by the program. It is essentially the total amount of virtual memory used by the process less memory used for instruction (`.text`) pages. It includes data pages used by shared libraries.

Resident Memory Size

This field displays the amount of resident memory associated with the process in units of kilobytes (KB).

CPU Time

This field displays the amount of CPU time used by the process in units of seconds.

User Time

This field displays the amount of time used on behalf of the process, excluding system time, in units of seconds.

System Time

This field displays the amount of system time used on behalf of the process in units of seconds.

Most Recent CPU

This field displays the CPU upon which the process last executed.

Affinity Mask

This field displays the CPU affinity mask which controls the list of CPUs upon which the process can execute. When the CPU affinity does not designate all CPUs on the system, the mask is displayed as a hexadecimal number. The least significant bit in the mask represents logical CPU 0.

Nice Value

This field displays the *nice* value as set by the **nice** command or the **nice(2)** system service. The nice value provides an initial base for the priority of processes in the **SCHED_OTHER** scheduling class.

Priority

This field displays the internal kernel priority value. Lower values represent more urgent priorities. For processes in the **SCHED_OTHER** scheduling class, the priority is adjusted by the kernel as the process runs, based on CPU utilization.

Real-time Priority

This field displays the real-time priority within the process's scheduling class.

Scheduling Class

This field displays the scheduling class associated with the process.

Command

This field displays the command name of the process in short or long form. Selection of short or long form is controlled on a per-frame basis using the Process Monitor Panel pop-up menu activated by clicking the third mouse button.

Process Monitor Drag and Drop Operations

Individual processes, threads, or groups of processes and threads can be dragged onto various destination panels and drop targets.

To drag an individual process or thread, middle-click anywhere on the row describing the process or thread and drag the pointer to the destination area and release the mouse button. Similarly, you can drag all the processes associated with a frame by middle-clicking on the folder icon of the frame and dragging the pointer to the destination area.

Dragging a multi-threaded process affects all threads associated with the process. Similarly, dragging a frame which contains multi-threaded processes affects all threads associated with those processes.

When dragging a process or thread, all selected processes and threads in the panel are included in the drag and drop operation. If no processes or threads are selected, the only item included in the drag and drop operation is the process or thread associated with the row indicated by the mouse pointer.

The Process Monitor panel supports the following drag and drop operations:

- Dragging a process, thread, or frame to a CPU box in the CPU Status panel binds the processes and threads to the corresponding CPU.

- Dragging a process or thread onto the Process Scheduler dialog changes the dialog to refer to that process or thread.
- Dragging a process, thread, or frame onto the Kill drop target causes the processes and threads to be killed with a **SIGKILL** signal.
- Dragging a process, thread, or frame onto the Unbind drop target causes the processes and threads to be unbound from any CPUs.

Process Monitor Pop-up Menu

While positioned in the Process Monitor panel over an open frame, right-clicking launches the Process Monitor pop-up menu.

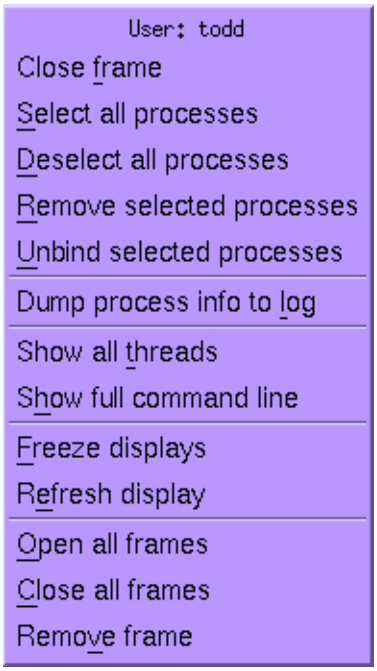

Figure 3-5. Process Monitor pop-up menu

The following paragraphs describe the menu items in more detail. All operations within the menu operate solely on the frame associated with the pop-up menu, unless otherwise noted below.

Close Frame

This menu item closes the frame, hiding all processes in the frame. The frame remains in the panel.

Open Frame

This menu item opens the frame, displaying all processes in the frame.

Select all processes

This menu item causes all processes in the frame to be selected.

Deselect all processes

This menu item deselects all processes in the frame.

Unbind selected processes

This menu item clears the CPU binding of the processes selected in the frame. The process's CPU affinity is set to *all CPUs*.

Dump process info to log

This menu item causes a textual description of all processes in the frame to be sent to the NightTune log file. If logging is not enabled, a dialog will inform the user. Logging can be enabled using the Logging item from the NightTune menu.

Show all threads

This menu item causes the list of threads to be expanded for all processes in the frame.

Hide all threads

This menu item causes the list of threads for all processes in the frame to be hidden, leaving a single process line that summarizes the activities of all the threads.

Show full command line

This menu item causes the expanded command name to be shown for all processes in the frame.

Show brief command line

This menu item causes the simplified command name to be shown for all process in the frame.

Freeze/UnFreeze Display

This menu item toggles the Freeze setting for this panel. When frozen, this panel is not refreshed automatically.

Refresh display

This menu item causes all process information for all frames in the Process Monitor panel to be immediately updated, regardless of the Freeze setting.

Open all frames or Close all frames

This menu item causes the respective action on all frames in the Process Monitor panel.

Remove Frame

This menu item removes the frame from the panel.

Process Scheduler

You can use the Process Scheduler dialog to alter the scheduling attributes and CPU affinity for any process for which you have appropriate privileges.

The Process Scheduler dialog is launched by clicking on the right-arrow icon on a process line in the Process Monitor panel.

The following illustrates the Process Scheduler dialog:

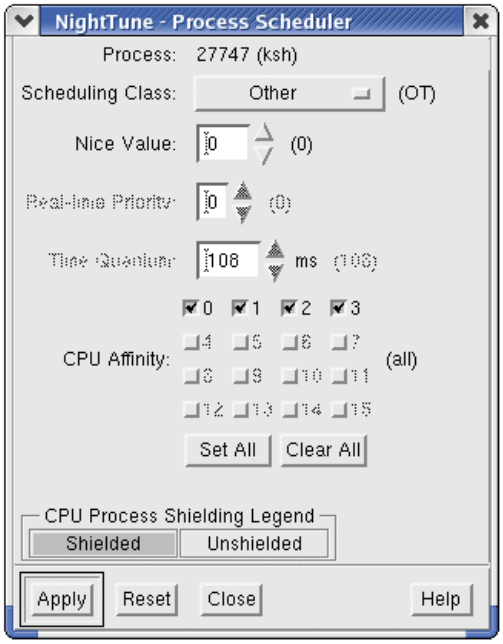

Figure 3-6. Process Scheduler dialog

Descriptions of the text fields and controls contained in the Process Scheduler dialog follow:

Process

Displays the process ID (PID) and simple name of the process currently referenced by the dialog. You can use this window as a drop target for processes; dropping a

process onto the window changes the dialog to refer to that process. You cannot drop multiple processes or a process with more than one thread on the dialog.

Scheduling Class

This menu item allows you to select one of the three scheduling classes supported by the operating system:

Other

Selects the **SCHED_OTHER** scheduling policy which is the default universal time-sharing policy used by most processes. The priority of the process is adjusted by the operating system based on CPU utilization. Processes using this policy will always have a less favorable priority than those in the remaining classes.

First In-First Out

Selects the **SCHED_FIFO** scheduling policy. The priority of processes within this policy are static — they are not adjusted by the operating system. Processes retain use of the CPU until they block, voluntarily yield the CPU, or are preempted by higher priority processes or interrupts.

Round Robin

Selects the **SCHED_RR** scheduling policy. This policy is identical to **SCHED_FIFO** except that processes are placed at the end of the run queue for their priority when their execution time-slice (time quantum) expires or when they voluntarily yield the CPU.

To the right of the option menu, the current scheduling class for the process is displayed using abbreviated notation: OT (Other), FF (First In-First Out), RR (Round Robin).

Nice Value

This text field allows you to specify a nice value for processes scheduled under the Other scheduling class.

To the right of the text field and step arrows, the current nice value for the process is displayed.

The Nice Value is only sensitized for processed using the Other scheduling class.

Real-Time Priority

This text field allows you to specify the priority within the associated real-time scheduling policy, Round Robin or First In-First Out.

Values for the priority must be in the range 1..99.

To the right of the text field and step arrows, the current real-time priority value for the process is displayed in parentheses.

The Real-Time Priority is desensitized for processes using the Other scheduling class.

Time Quantum

This text field allows you to specify the duration of the execution time-slice for processes using the Round Robin scheduling class. It is not applicable to any other scheduling class.

The units of time are in milliseconds.

To the right of the text field and step arrows, the current time quantum for the process is displayed within parentheses.

The Time Quantum is only sensitized for processes using the Round Robin scheduling class.

The operating system only supports certain step-values for the Time Quantum; if you enter a value by hand, it will be rounded to the nearest appropriate quantum value.

CPU Affinity

The CPU Affinity area allows you to specify individual CPUs where the process is allowed to execute. If a single CPU checkbox is checked, the process is bound to that CPU and the process is displayed in the Processes list in the associated CPU box in the CPU Status panel.

For each CPU that has been shielded from processes, a darkened gray background appears around the checkbox. If a CPU has been shielded from processes, no processes will execute on that CPU unless they are specifically bound to that CPU (and no others).

To the right of the CPU checkboxes, the current CPU affinity mask is displayed as a hexadecimal value in parentheses. The least significant bit represents logical CPU 0.

For more information about CPU affinity and scheduling policies and priorities, refer to the **`mpadvise(2)`**, **`sched_setaffinity(2)`**, **`sched_setscheduler(2)`**, and **`sched_setparam(2)`** manual pages. Additional information is available in the *RedHawk Linux User's Guide*.

Process Scheduling Operations

The process scheduling operations are controlled using the buttons at the bottom of the dialog:

Apply

Clicking Apply applies any scheduling changes made in the dialog. You may not see a change reflected immediately since the operating system may defer certain operations until the next time the process becomes active. This is particularly likely if a process is being starved of CPU time; in this case try unbinding the process, or binding it to a different CPU, and making the change again.

Reset

Clicking Reset causes the current process scheduling attributes to be reflected in the dialog, discarding any changes that have not yet been applied.

Close

Clicking Close closes the dialog. Any changes that have not been applied are discarded.

Help

Clicking Help presents this section of the manual in the HyperHelp viewer.

CPU Status Panel

The CPU Status panel describes the status of each CPU on the system and provides shielding controls.

The following illustrates the CPU Status panel:

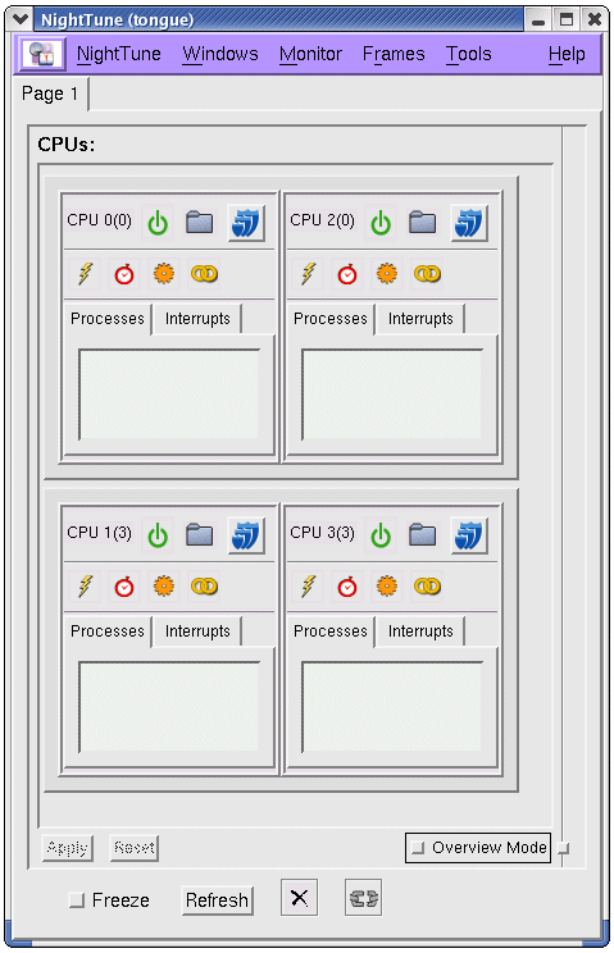

Figure 3-7. CPU Status panel

Overview Mode checkbox

The Overview Mode checkbox at the bottom of the panel controls the content of the CPU boxes. When checked, the list of bound Processes and Interrupts disappears from each CPU box to save space. Clearing the checkbox causes the lists to be displayed.

In Overview Mode, the lists of processes, threads, and interrupts are replaced by text labels indicating the number of processes, threads, and interrupts bound to each CPU, respectively.

CPU Boxes

A single CPU box is displayed for each CPU on the sytem. CPUs which are hyper-threaded siblings are grouped together horizontally.

CPU Box

Each CPU box contains the following labels, lists, and controls:

CPU Label

The logical CPU number is displayed as well as the physical CPU number in parentheses.

On dual-core systems, three values are displayed:

```
CPU l(c)[r]
```

where:

- ***l*** is the logical CPU number
- ***c*** is the chip number
- ***r*** is the core number

On such systems, each chip contains two core processors and each core can be hyperthreaded; in total, up to four software threads can be handled simultaneously.

Active Toggle 

The Active icon indicates whether the CPU is marked Down (black) or is Active (green) and available for processing. CPUs may be marked Down so that they do not interfere with processing on their sibling hyper-threaded CPU.

Folder Toggle 

Clicking the Folder icon to open the folder causes a new CPU frame to open in Process Monitor panel, listing all processes bound to that CPU. Subsequently clicking the open folder icon closes the folder and removes the CPU frame from the Process Monitor panel.

Maximum Shielding Icon 

The Maximum Shielding icon is provided as a convenience. When clicked, it causes the following changes to the CPU Status panel:

- the Interrupt Shielding toggle is set to the shielded state for the CPU
- the Local Timer Shielding toggle is set to the shielded state for the CPU

- the Process Shielding toggle is set to the shielded state for the CPU
- the Hyper-Threading Shielding toggle is set to the shielded state for the CPU (which implies that the hyper-threaded sibling CPU's Active toggle is set to the CPU Down state, if it exists)

Once these settings are applied, the CPU will attain the maximum shielded state and will remain unused except for processes and interrupts specifically bound to that CPU.

Interrupt Shielding Toggle 

The Interrupt Shielding toggle indicates whether the CPU will be shielded from interrupts. Note that the Local Timer interrupt is not affected by this setting. When shielded, the only interrupts allowed on this CPU are those whose CPU affinity has been specifically bound to this CPU.

Local Timer Shielding Toggle 

The Local Timer Shielding toggle indicates whether the CPU will be shielded from the Local Timer interrupt.

Process Shielding Toggle 

The Process Shielding toggle indicates whether the CPU will be shielded from processes. When shielded, the only processes allowed to execute on the CPU are those whose CPU affinity has been specifically bound to this CPU. This includes kernel daemons that run in user mode that provide post-interrupt processing.

Hyper-Threading Shielding Toggle 

The Hyper-Threading Shielding toggle indicates whether the hyper-threaded sibling CPU is marked Active or Down. When shielded, the Active toggle for the sibling is set to the Down state.

Processes

The Processes list area includes the process ID and process name of all processes bound to this CPU. The Processes list area is not visible when the Overview Mode checkbox is selected.

When a process has multiple threads, the process ID displayed in this list area is annotated with a pair of numbers representing the number of threads bound to the CPU and the number of threads in the entire process, respectively. It is possible for the same process ID to be present in two different CPUs' Processes lists if some threads of the process are bound to one CPU and others are bound to another. Clicking the Folder icon opens a CPU frame in the Process Monitor panel which details which threads are bound to the CPU.

Double-clicking a process causes the Process Scheduler dialog to refer to the associated process -- the dialog is launched if it is not already present.

Processes in the list can be dragged to other CPUs and can be unbound by dragging them to the Unbind drop target (see "Drop Targets" on page 2-12) or using the CPU pop-up menu (see "CPU Pop-up Menu" on page 3-22 and "CPU Drag and Drop Operations" on page 3-21).

Note that no processes or interrupts appear in this panel if the system has only a single CPU.

Interrupts

The Interrupts list area includes the IRQ number and interrupt name of all interrupts bound to this CPU. The Interrupts list area is not visible when the Overview Mode checkbox is selected.

Double-clicking an interrupt causes the Interrupt Affinity dialog to refer to the associated process -- the dialog is launched if it is not already present.

Interrupts in the list can be dragged to other CPUs and can be unbound by dragging them to the Unbind drop target (see "Drop Targets" on page 2-12) or using the CPU pop-up menu (see "CPU Pop-up Menu" on page 3-22 and "CPU Drag and Drop Operations" on page 3-21).

NightTune does not allow certain combinations of settings. There must be at least one active CPU that is not shielded from interrupts and at least one CPU that is not shielded from processes.

Once a change to a shielding toggle has been made, the CPU Status panel stops refreshing until the changes are applied or discarded.

Changes to the shielding toggles and buttons are not applied until clicking the Apply button at the bottom of the panel.

CPU Shielding Operations

The CPU shielding operations are controlled using the buttons at the bottom of the panel:

Apply

Clicking Apply attempts to apply the shielding changes to the system. The application may fail if the user does not have sufficient capabilities; see "Capabilities" on page 1-2 for more information. Additionally, the shielding changes may conflict with current CPU usage. Attempting to set a CPU's status to Down will be rejected if processes or interrupts are currently bound to the CPU. A diagnostic will be displayed if the shielding operation fails.

Reset

Clicking Reset restores the CPU Status panel display with current information, discarding any changes that have not yet been applied.

CPU Drag and Drop Operations

Individual interrupts, processes or groups of interrupts or processes can be dragged onto various destination panels and drop targets.

To drag a process or interrupt, select the items of interest in the Processes or Interrupts area using the left mouse button. Using the middle mouse button, click in the Processes or Interrupts area and drag the pointer to the destination area and release. Alternatively, if no processes or interrupts are selected in the area, click the middle mouse button anywhere on the row which describes the process or interrupt and drag to the destination area and release.

The CPU Status panel supports the following drag and drop operations:

- Dragging processes onto a CPU box in the CPU Status panel binds the processes to the corresponding CPU. When dragging a multi-threaded process (identified by a numeric pair enclosed in parentheses following the process name) from a CPU box to another CPU box, only the threads bound to the source CPU are moved.
- Dragging processes onto the Process Monitor panel adds the processes to the list of processing being monitored.
- Dragging a process onto the Process Scheduler dialog panel changes the dialog to refer to that process. Dragging multiple processes or multi-threaded processes to the Process Scheduler dialog is not supported.
- Dragging interrupts onto a CPU box in the CPU Status panel binds the interrupts to the corresponding CPU.
- Dragging an interrupt onto the Interrupt Affinity dialog of the Interrupts panel changes the dialog to refer to that interrupt. Dragging multiple interrupts to the Interrupt Affinity dialog is not supported.
- Dragging processes onto the Kill drop target causes the processes to be killed with a **`SIGKILL`** signal.
- Dragging processes or interrupts onto the Unbind drop target causes the processes or interrupts to be unbound from any CPUs.

Note that interrupts can be dragged only if NightTune is running as `root` or if the user running NightTune has been granted the proper capabilities (see “Capabilities” on page 1-2).

CPU Pop-up Menu

While positioned over a CPU box in the CPU Status panel, right-clicking launches a CPU pop-up menu for the associated CPU.

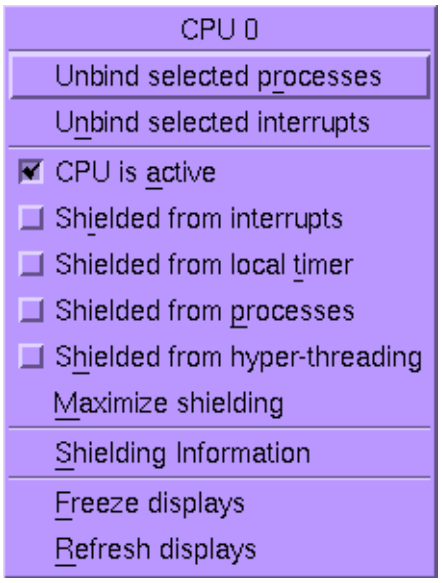

Figure 3-8. CPU pop-up menu

All activities in the CPU pop-up menu apply solely to the associated CPU, except where noted in the detailed descriptions below:

Unbind selected processes

This menu item immediately changes the CPU affinity of the selected processes in the Processes list to include all CPUs; effectively unbinding them from the CPU. This item is not available on a system with only a single CPU.

Unbind selected interrupts

This menu item immediately changes the CPU affinity of the selected interrupts in the Interrupts list to include all CPUs; effectively unbinding them from the CPU. This item is not available on a system with only a single CPU.

Note that interrupts can be unbound only if NightTune is running as `root` or if the user running NightTune has been granted the proper capabilities (see "Capabilities" on page 1-2).

CPU is active

This menu item toggles the state of CPU Active. When a CPU is Active, it is available for processes and interrupts. When a CPU is Inactive, it is unavailable.

Making a CPU inactive is typically only useful when hyper-threading is enabled. It prevents unpredictable interference with the hyper-threaded sibling CPU.

Shielded from interrupts

This menu item toggles the state of Interrupt Shielding.

Shielded from local timer

This menu item toggles the state of Local Timer Shielding.

Shielded from processes

This menu item toggles the state of Process Shielding.

Shielded from hyper-threading

This menu item toggles the state of Hyper-threading Shielding.

Maximize shielding

This menu item sets the state of all shielding toggles to shielded and sets the Active toggle for the sibling hyper-threaded CPU to Down, if it is hyper-threaded.

A CPU in the maximum shielded state will remain unused except for processes and interrupts specifically bound to that CPU.

Shielding Information...

This menu item pops up a textual description of the state of shielding as described in the current CPU box. If changes have been made to the shielding icons, but have not yet been applied, the description refers to the changes that may be applied. Otherwise, the description refers to the current state of the CPU.

See the *RedHawk Linux Real-Time User's Guide* for more information on CPU shielding and process and interrupt CPU affinity.

Freeze/UnFreeze Display

This menu item toggles the Freeze setting for this panel. When frozen, this panel is not refreshed automatically.

Refresh displays

This menu item causes all process information for all frames in the Process Monitor panel to be immediately updated, regardless of the Freeze setting.

Interrupt Activity Panel

The Interrupt Activity panel provides three panes that display interrupt activity on the system: the Interrupt Table, the Interrupt Bar Graphs, and the Interrupt Line Graphs.

Interrupt Table

The following illustrates the Interrupt Table:

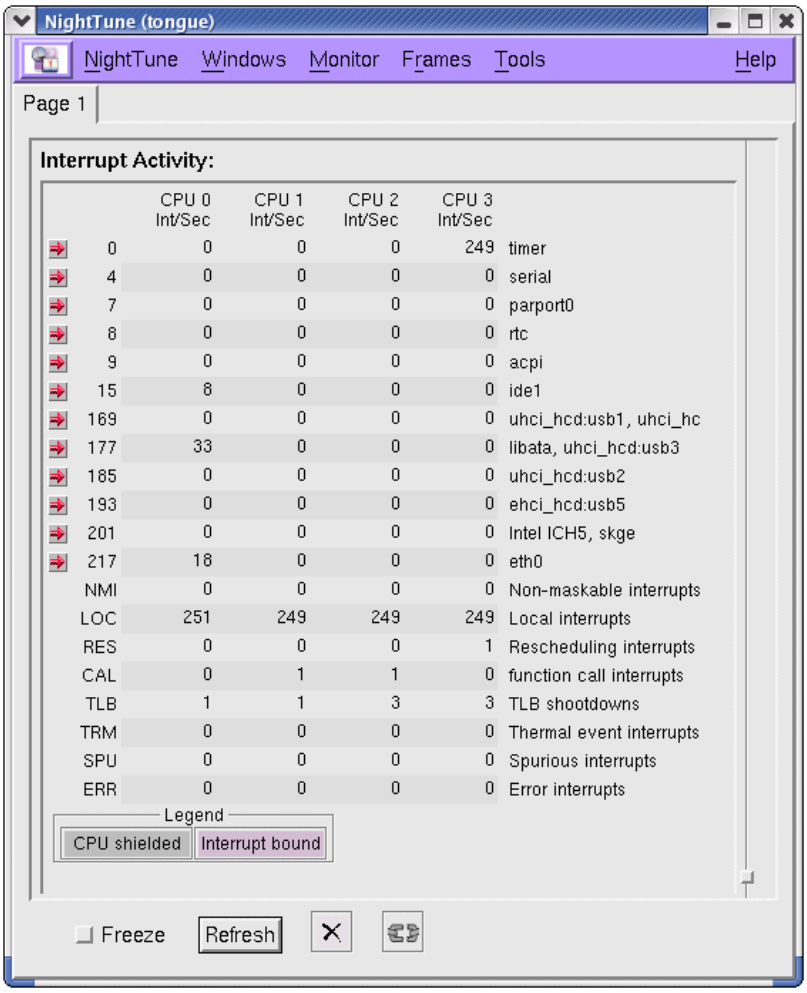

Figure 3-9. Interrupt Activity panel

The Interrupt Table provides information on individual interrupts and allows the user to change an interrupt's CPU affinity.

The information displayed in this area includes:

Control Button

For each interrupt whose affinity can be controlled, a push-button with a red arrow icon is provided. Clicking on the push-button launches the Interrupt Affinity dialog which allows you to change the CPU affinity for that interrupt. If the user lacks the capabilities to change the CPU affinity, the Control Button button will not appear. See "Capabilities" on page 1-2 for more information.

Interrupt Value

The IRQ value or mnemonic classification for the interrupt is displayed.

Interrupts Per Second

For each CPU, the number of interrupts per second is displayed. For interrupts which have a CPU affinity which does not include all CPUs, a purple background surrounds all interrupt-per-second values for that interrupt in columns associated with CPUs in the CPU affinity mask. Thus, at a glance you can determine which interrupts are directed or bound to which CPUs.

For CPUs which are shielded from interrupts, the column header for that CPU will have a dark grey background.

Interrupt Description

For interrupts associated with devices, the device name is displayed. For other interrupts, a short functional description of their purpose is displayed.

The interrupt data is automatically refreshed at a cyclic rate. The refresh interval can be adjusted using the Preferences dialog launched from the Preferences… option of the NightTune menu.

Interrupt Control Drag and Drop Operations

Individual interrupts can be dragged onto various destination panels and drop targets.

To drag an interrupt, middle-click anywhere on the interrupt row and drag the pointer to the destination area and release.

- Dragging an interrupt onto a CPU box in the CPU Status panel binds the interrupt to the corresponding CPU.
- Dragging an interrupt onto the Interrupt Affinity dialog causes the dialog to refer to the interrupt.
- Dragging an interrupt onto the Unbind drop target causes the interrupt to be unbound from any CPUs.

Interrupt Affinity Dialog

The Interrupt Affinity dialog displays the current CPU affinity of a specific interrupt and allows you to change it.

The following illustrates the Interrupt Affinity dialog:

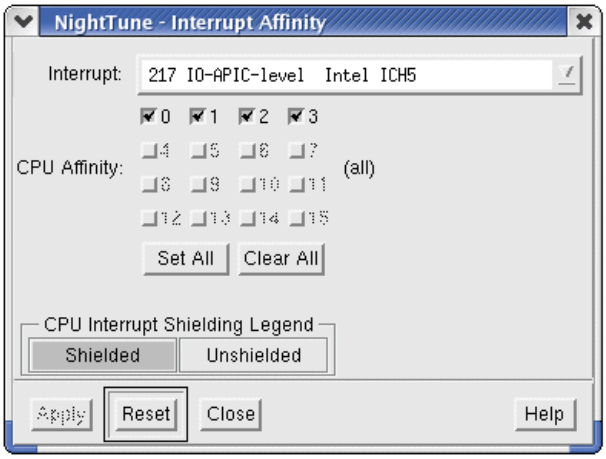

Figure 3-10. Interrupt Affinity dialog

The dialog consists of the following functional labels and controls:

Interrupt Selection

The IRQ value and interrupt name currently associated with the dialog is displayed. A drop-down list allows you to select a different interrupt.

The Interrupt Affinity dialog supports drag and drop operations from the Interrupt Table and from the Interrupts list in the CPU Status panel; an interrupt dragged to the dialog changes the dialog to select the interrupt which is the source of the drag operation.

CPU Affinity Checkboxes

The CPU Affinity area allows you to specify individual CPUs where the interrupt can be delivered. If a single CPU checkbox is checked, the interrupt will be bound to that CPU and the interrupt will be displayed in the Interrupts list in the associated CPU box in the CPU Status panel.

For each CPU that has been shielded from interrupts, a darkened grey border will appear around the checkbox. If a CPU has been shielded from interrupts, no interrupts will be delivered to that CPU unless they are specifically bound solely to that CPU.

To the right of the CPU checkboxes, the current CPU affinity mask is displayed as a hexadecimal value in parentheses. The least significant bit represents logical CPU 0.

Apply

Clicking the Apply button applies the CPU affinity selected in the dialog to the interrupt. If the user lacks the capabilities to change the CPU affinity, the Apply button will be desensitized. See “Capabilities” on page 1-2 for more information.

Reset

Clicking the Reset button discards any changes not yet applied and refreshes the displayed values to the current affinity of the interrupt.

Close

Clicking the Close button closes the dialog; any changes made to the dialog that have not been applied are discarded.

Help

Clicking the Help button displays this section in the on-line HyperHelp viewer.

Interrupt Bar Graphs

The Interrupt Bar Graphs provide individual bar graphs showing the number of interrupts per second per CPU.

The following illustrates the Interrupt Bar Graphs:

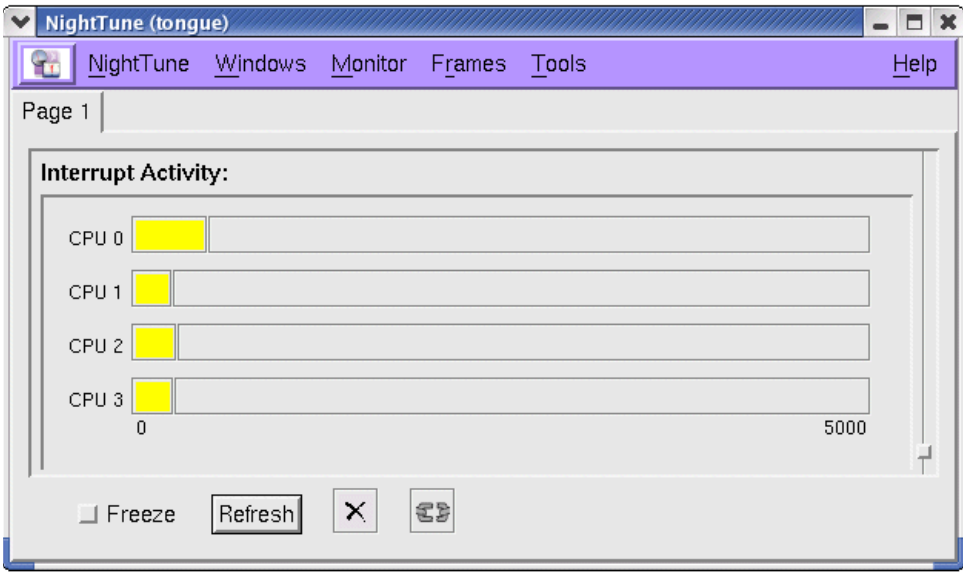

Figure 3-11. Interrupt Activity Bar Graphs

Regardless of which interrupts are displayed in the Interrupt Table and Interrupt Line Graphs, these bar graphs represent **all** interrupt activity on the system.

The displayed information is automatically refreshed at a cyclic rate. The refresh interval can be adjusted using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Interrupt graphs are automatically scaled by NightTune based on peak interrupt activity as monitoring continues. You can cause NightTune to rescale the graphs based on current interrupt activity using the Rescale displays option from the Interrupt Activity pop-up menu.

Interrupt Line Graphs

The Interrupt Line Graphs provide individual line graphs for each interrupt for each CPU.

The following illustrates the Interrupt Line Graphs:

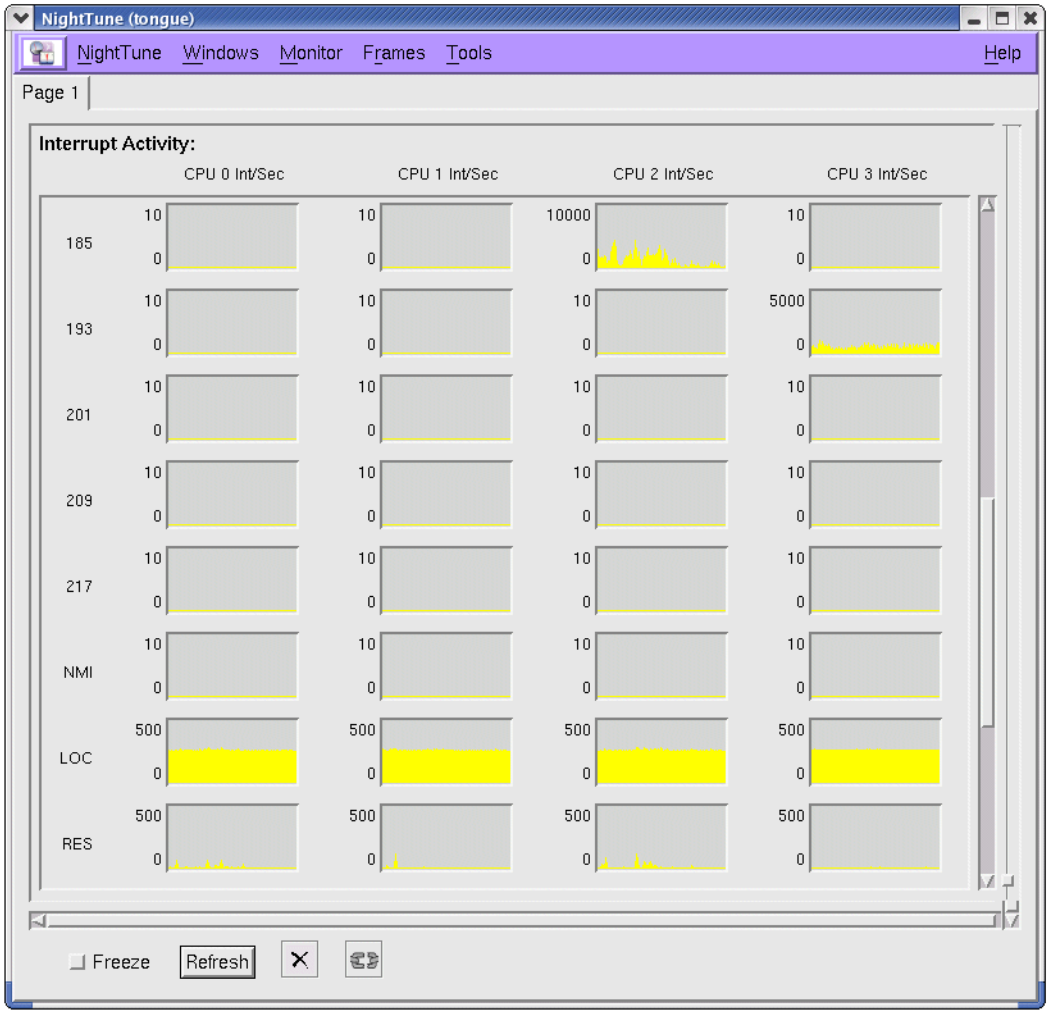

Figure 3-12. Interrupt Line Graphs

The number of interrupts per second is displayed in each graph. The graphs automatically scroll as the information is cyclically refreshed.

The refresh interval can be changed using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Interrupt graphs are automatically scaled by NightTune based on peak interrupt activity as monitoring continues. You can cause NightTune to rescale the graphs based on current interrupt activity using the Rescale displays option from the Interrupt Activity pop-up menu.

Interrupt Activity Pop-up Menu

While positioned in the Interrupt Activity panel, right-clicking launches the Interrupt Activity pop-up menu.

Figure 3-13. Interrupt Activity pop-up menu

The following paragraphs describe the menu items in detail:

Display Interrupts...

This menu item launches the Interrupt Selection dialog which allows you to select which interrupts will be displayed in the Interrupt Table and Interrupt Line Graphs panes in the panel. This menu item does not affect the interrupt activity as shown in the Interrupt Bar Graphs.

Show/Hide text display

This menu item toggles the visibility of the Interrupt Table within the panel.

Show/Hide bargraph display

This menu item toggles the visibility of the Interrupt Bar Graphs within the panel.

Show/Hide linegraph display

This menu item toggles the visibility of the Interrupt Line Graphs within the panel.

Freeze/UnFreeze displays

This menu item toggles the Freeze setting for the Interrupt Activity panel. When frozen, data values are not refreshed automatically. This menu item overrides the Freeze setting for the window, but only applies to displays within the Interrupt Activity panel.

Refresh displays

This menu item causes all data in all displays within the Interrupt Activity panel to be refreshed once, regardless of the Freeze setting.

Rescale displays

This menu item causes NightTune to rescale all graphs within the Interrupt Activity panel based on current interrupt activity.

Processor Usage Panel

The Processor Usage panel provides three panes that display the CPU usage of the system: the CPU Table, the CPU Bar Graphs, and the CPU Line Graphs.

The percentages displayed reflect CPU activity over the period of time defined by the refresh interval.

Processor Table

The following illustrates the Processor Table:

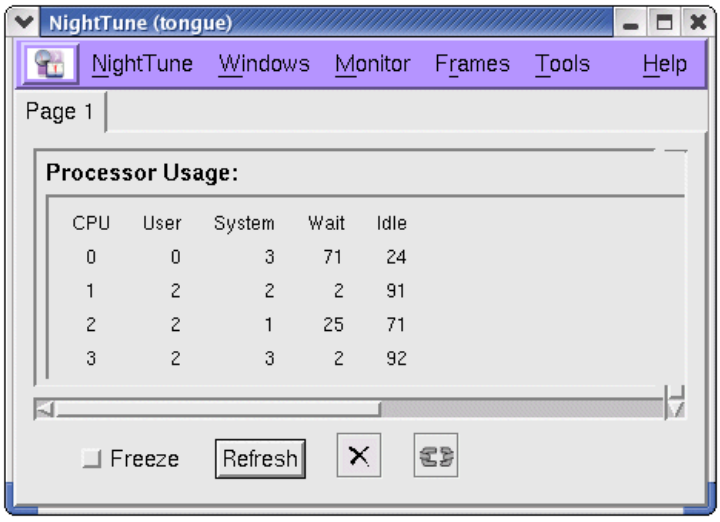

Figure 3-14. Processor Usage Table

The Processor Usage Table provides information on CPU usage for each CPU in the system.

The information displayed in this area includes:

CPU

The logical CPU number is displayed in this column.

User

The percentage of time the CPU was executing in user mode is displayed in this column. This excludes kernel execution but does include execution of kernel daemons which handle post-interrupt processing.

System

The percentage of time the CPU was executing in the operating system kernel is displayed in this column. This includes time spent executing system service calls on behalf of user processes as well as interrupt and machine exception processing.

Wait

The percentage of time the CPU was idle with at least one process waiting for an I/O operation (on any CPU) to complete is displayed in this column.

Idle

The percentage of time the CPU was executing the idle loop is displayed in this column.

The processor usage data is automatically refreshed at a cyclic rate. The refresh interval can be adjusted using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Processor Bar Graphs

The Processor Bar Graphs provide show the percentage of CPU time used for each CPU.

The following illustrates the Processor Bar Graphs:

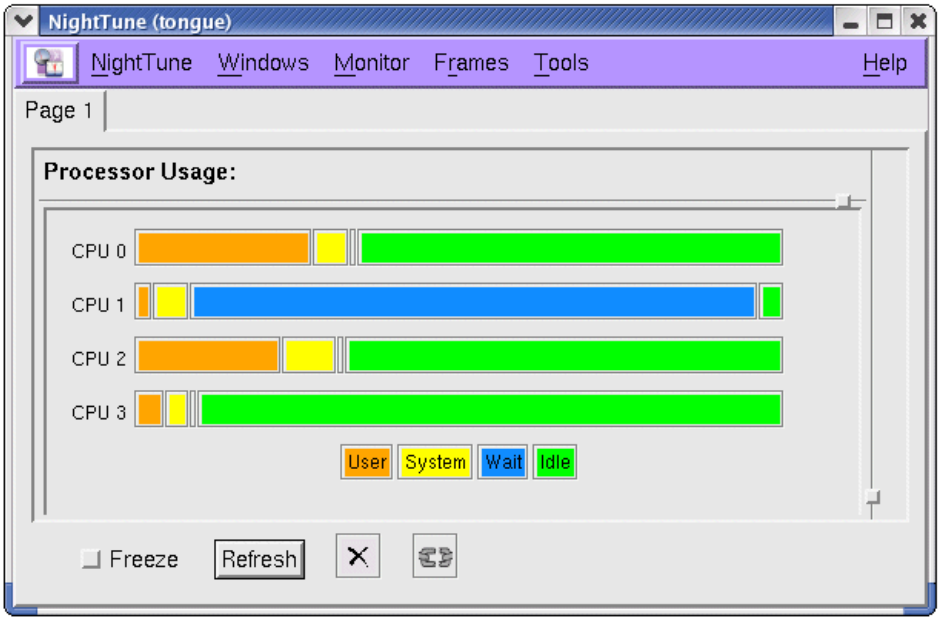

Figure 3-15. Processor Usage Bar Graphs

For each CPU, the User, System, Wait, and Idle times are shown as percentages of CPU execution using color-coded, horizontal bars.

See “Processor Table” on page 3-31 for definitions of User, System, Wait and Idle times.

Processor Line Graphs

The Processor Line Graphs provide individual line graphs for User, System, Wait, and Idle times, expressed as a percentage of CPU execution for each CPU.

The following illustrates the Processor Line Graphs:

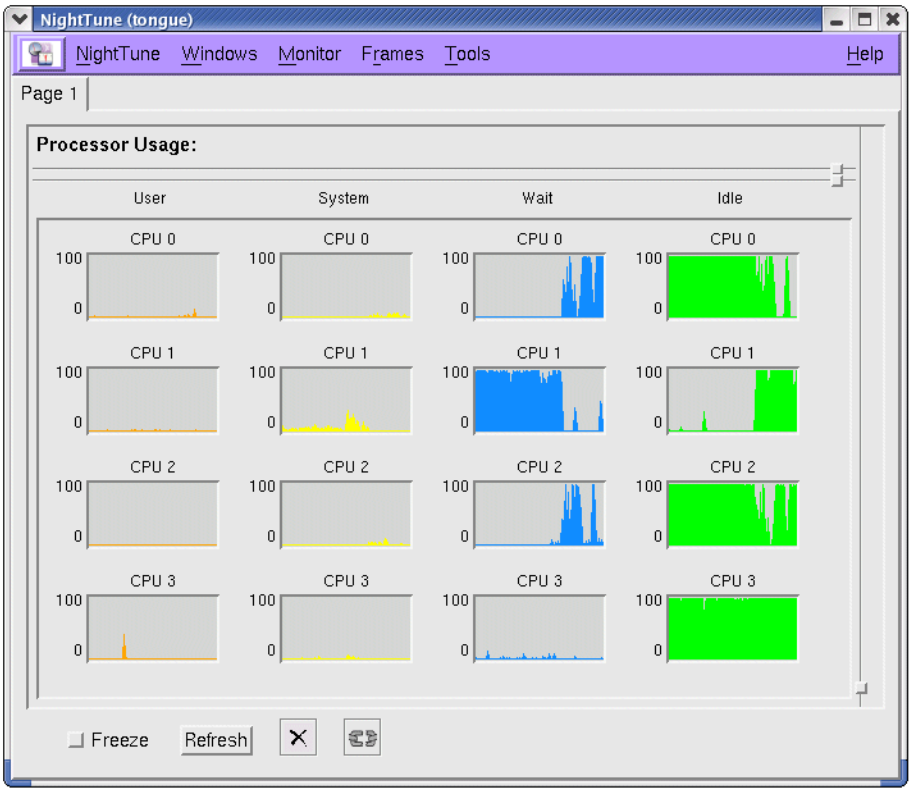

Figure 3-16. Processor Line Graphs

See "Processor Table" on page 3-31 for the definitions of User, System, Wait, and Idle times.

Processor Usage Pop-up Menu

While positioned in the Processor Usage panel, right-clicking launches the Processor Usage pop-up menu.

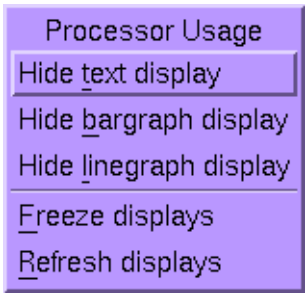

Figure 3-17. Processor Usage pop-up menu

The following paragraphs describe the menu items in detail:

Show/Hide text display

This menu item toggles the visibility of the Processor Table within the panel.

Show/Hide bargraph display

This menu item toggles the visibility of the Processor Bar Graphs within the panel.

Show/Hide linegraph display

This menu item toggles the visibility of the Processor Line Graphs within the panel.

Freeze/UnFreeze displays

This menu item toggles the Freeze setting for the Processor Usage panel. When frozen, data values are not refreshed automatically. This menu item overrides the Freeze setting for the window, but only applies to displays within the Processor Usage panel.

Refresh displays

This menu item causes all data in all displays within the Processor Usage panel to be refreshed once, regardless of the Freeze setting.

Context Switches Panel

The Context Switches panel provides three panes that display context switches on the system: the Context Switch Table, the Context Switch Bar Graphs, and the Context Switch Line Graphs.

Each table and graph displays the number of context switches per second per CPU that occurred over the period of time defined by the refresh interval.

Context Switch Table

The following illustrates the Context Switch Table:

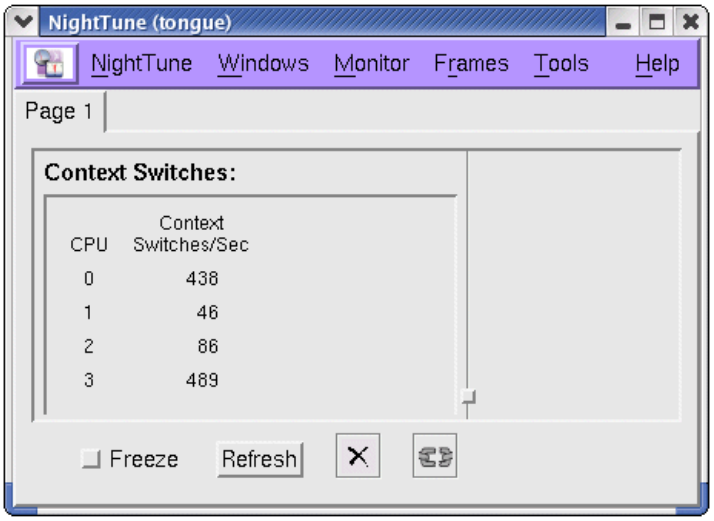

Figure 3-18. Context Switch Table

The information displayed in this area includes:

CPU Number

The logical CPU number is displayed in this column.

Context Switch Rate

The number of context switches per second is displayed in this column.

The context switch data is automatically refreshed at a cyclic rate. The refresh interval can be adjusted using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Context Switch Bar Graphs

The Context Switch Bar Graphs provide individual bar graphs for each CPU.

The following illustrates the Context Switch Bar Graphs:

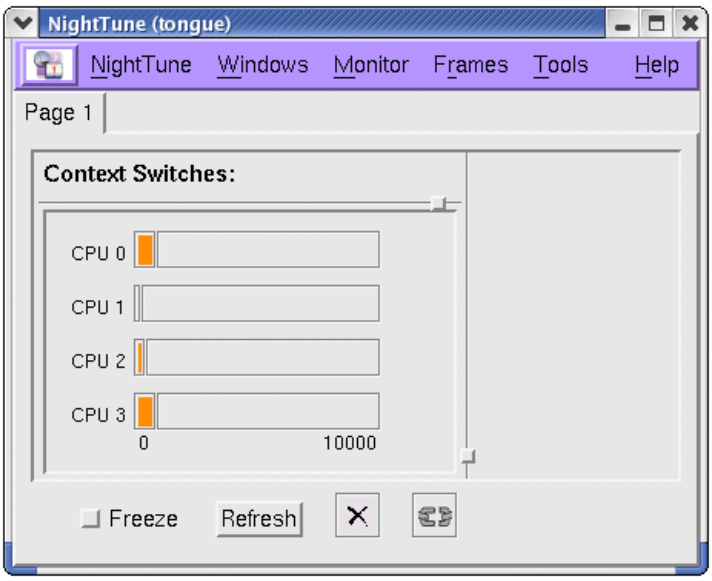

Figure 3-19. Context Switch Bar Graphs

The number of context switches per second is displayed as a horizontal bar for each CPU.

The displayed information is automatically refreshed at a cyclic rate. The refresh interval can be adjusted using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Context switch graphs are automatically scaled by NightTune based on peak context switch activity as monitoring continues. You can cause NightTune to rescale the graphs based on the current context switch rate using the Rescale displays option from the Context Switches pop-up menu.

Context Switch Line Graphs

The Context Switch Line Graphs provide individual line graphs detailing the context switch rate for each CPU.

The following illustrates the Context Switch Line Graphs:

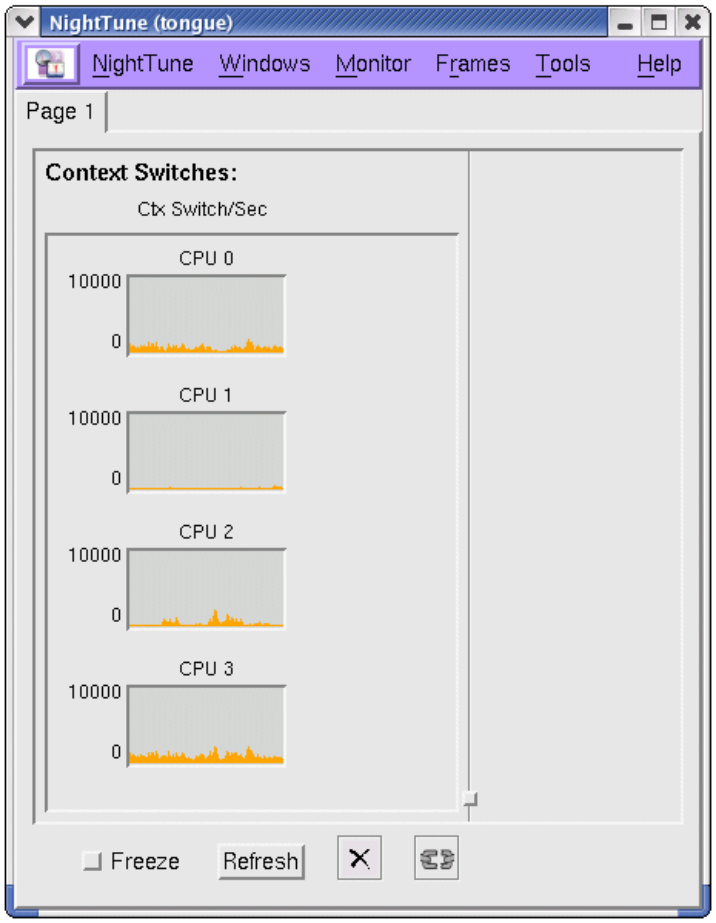

Figure 3-20. Context Switch Line Graphs

The number of context switches per second for a specific CPU is displayed as a vertical line in each graph. The graphs automatically scroll as the information is cyclically refreshed.

The refresh interval can be changed using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Context switch graphs are automatically scaled by NightTune based on peak context switch activity as monitoring continues. You can cause NightTune to rescale the graphs based on the current context switch rate using the Rescale displays option from the Context Switches pop-up menu.

Context Switches Pop-up Menu

While positioned in the Context Switches panel, right-clicking launches the Context Switches pop-up menu.

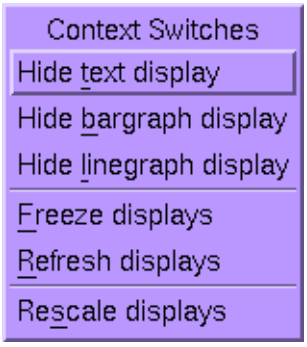

Figure 3-21. Context Switches pop-up menu

The following paragraphs describe the menu items in detail:

Show/Hide text display

This menu item toggles the visibility of the Context Switch Table within the panel.

Show/Hide bargraph display

This menu item toggles the visibility of the Context Switch Bar Graphs within the panel.

Show/Hide linegraph display

This menu item toggles the visibility of the Context Switch Line Graphs within the panel.

Freeze/Unfreeze displays

This menu item toggles the Freeze setting for the Context Switches panel. When frozen, data values are not refreshed automatically. This menu item overrides the Freeze setting for the window, but only applies to displays within the Context Switches panel.

Refresh displays

This menu item causes all data in all displays within the Context Switches panel to be refreshed once, regardless of the Freeze setting.

Rescale displays

This menu item causes NightTune to rescale all graphs within the Context Switches panel based on the current context switch rate.

Virtual Memory Activity Panel

The Virtual Memory Activity panel provides two panes that display physical memory page transfer rates on the system: the Page Transfer Table and the Page Transfer Line Graphs.

Each table and graph displays the number of physical memory page transfers per second that occurred over the period of time defined by the refresh interval.

Page Transfer Table

The following illustrates the Page Transfer Table:

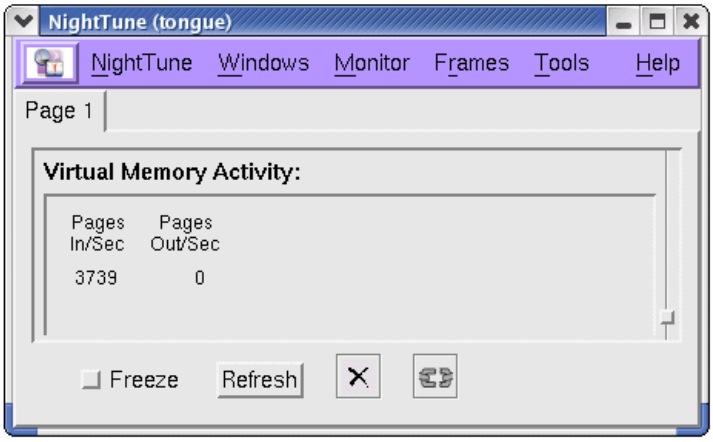

Figure 3-22. Page Transfer Table

The information displayed in this area includes:

Page-In Transfers

The number of physical memory page transfers due to page-in operations per second is displayed in this column.

Page-Out Transfers

The number of physical memory page transfers due to page-out operations per second is displayed in this column.

The paging rate data is automatically refreshed at a cyclic rate. The refresh interval can be adjusted using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Page Transfer Line Graphs

The Page Transfer Line Graphs provide individual line graphs detailing the paging rates for Page-In and Page-Out operations.

The following illustrates the Page Transfer Line Graphs:

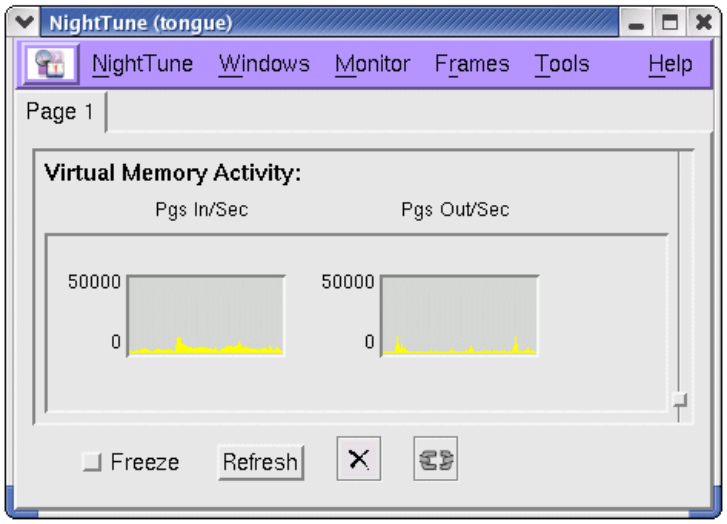

Figure 3-23. Page Transfer Line Graphs

The number of physical memory page transfers per second for each page operation type is displayed as a vertical line in each graph. The graphs automatically scroll as the information is cyclically refreshed.

The refresh interval can be changed using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Virtual memory activity graphs are automatically scaled by NightTune based on peak paging rates as monitoring continues. You can cause NightTune to rescale the graphs based on the current paging rate using the Rescale displays option from the Virtual Memory Activity pop-up menu.

Virtual Memory Activity Pop-up Menu

While positioned in the Virtual Memory Activity panel, right-clicking launches the Virtual Memory Activity pop-up menu.

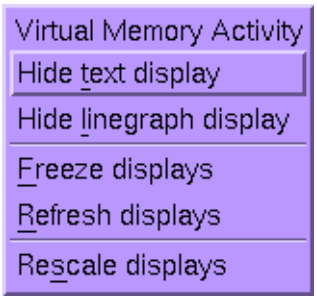

Figure 3-24. Virtual Memory Activity pop-up menu

The following paragraphs describe the menu items in detail:

Show/Hide text display

This menu item toggles the visibility of the Page Transfer Table within the panel.

Show/Hide linegraph display

This menu item toggles the visibility of the Page Transfer Line Graphs within the panel.

Freeze/Unfreeze displays

This menu item toggles the Freeze setting for the Virtual Memory Activity panel. When frozen, data values are not refreshed automatically. This menu item overrides the Freeze setting for the window, but only applies to displays within the Virtual Memory Activity panel.

Refresh displays

This menu item causes all data in all displays within the Virtual Memory Activity panel to be refreshed once, regardless of the Freeze setting.

Rescale displays

This menu item causes NightTune to rescale all graphs within the Virtual Memory Activity panel based on the current page transfer rate.

Disk Activity Panel

The Disk Activity panel provides two panes that display disk operation rates on the system: the Disk Operations Table and the Disk Operations Line Graphs.

Each table and graph displays statistics related to disk operations that occurred over the period of time defined by the refresh interval. Statistics for individual disks are displayed.

Disk Operations Table

The following illustrates the Disk Operations Table:

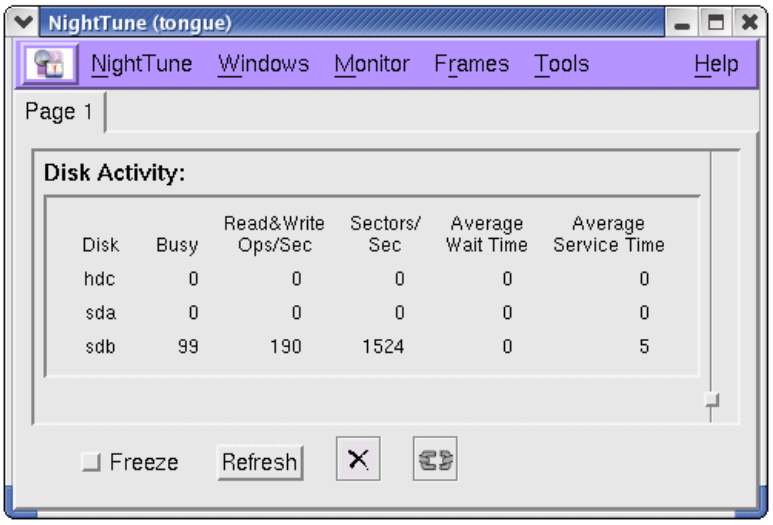

Figure 3-25. Disk Operations Table

The information displayed in this area includes:

Disk ID

The disk device name is displayed in this column.

Busy Percentage

The percentage of time the disk was busy servicing a transfer request is displayed in this column.

Read & Write Operations

The number of read and write transfer requests serviced by the disk per second is displayed in this column.

Sector Transfers

The number of sectors transferred to or from the disk per second is displayed in this column.

Average Wait Time

The average wait time, in milliseconds, that transfer requests waited on the queue before being serviced is displayed in this column.

Average Service Time

The average service transfer time, in milliseconds, is displayed in this column. The time includes seek time, rotational latency, and data transfer time.

The disk activity data is automatically refreshed at a cyclic rate. The refresh interval can be adjusted using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Disk Operations Line Graphs

The Disk Operations Line Graphs provide individual line graphs detailing metrics related to disk operations.

The following illustrates the Disk Operations Line Graphs:

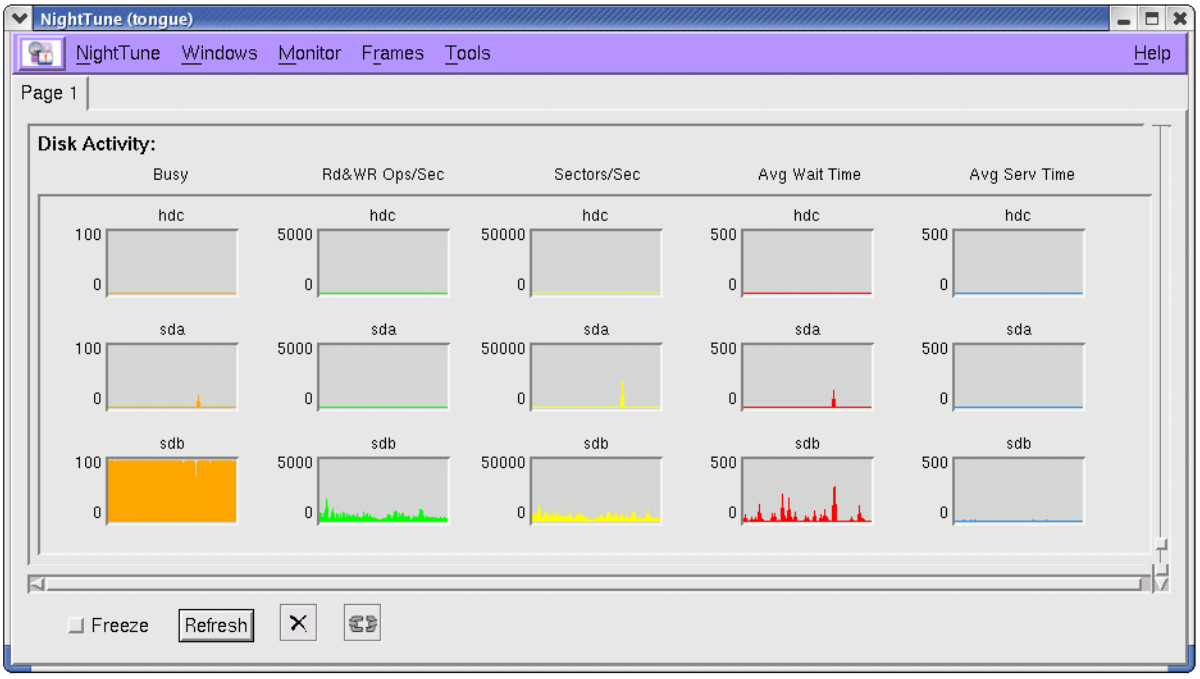

Figure 3-26. Disk Operations Line Graphs

Individual line graphs are shown for each disk for each metric. The value of the metric is displayed vertically in each graph. The metrics include: Busy Percentage, Read & Write Operation Rate, Sector Transferred Rate, Average Wait Time, and Average Service Time. See “Disk Operations Table” on page 3-43 for definitions of these metrics.

The refresh interval can be changed using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Disk activity graphs are automatically scaled by NightTune based on peak disk activity as monitoring continues. You can cause NightTune to rescale the graphs based on current disk activity using the Rescale displays option from the Disk Activity pop-up menu.

Disk Activity Pop-up Menu

While positioned in the Disk Activity panel, right-clicking launches the Disk Activity pop-up menu.

Figure 3-27. Disk Activity pop-up menu

The following paragraphs describe the menu items in detail:

Show/Hide text display

This menu item toggles the visibility of the Disk Operations Table within the panel.

Show/Hide linegraph display

This menu item toggles the visibility of the Disk Operations Line Graphs within the panel.

Freeze/Unfreeze displays

This menu item toggles the Freeze setting for the Disk Activity panel. When frozen, data values are not refreshed automatically. This menu item overrides the Freeze setting for the window, but only applies to displays within the Disk Activity panel.

Refresh displays

This menu item causes all data in all displays within the Disk Activity panel to be refreshed once, regardless of the Freeze setting.

Rescale displays

This menu item causes NightTune to rescale all graphs within the Disk Activity panel based on current disk activity.

Network Activity Panel

The Network Activity panel provides two panes that display network statistics on the system: the Network Activity Table and the Network Activity Line Graphs.

Each table and graph displays statistics related to network operations that occurred over the period of time defined by the refresh interval. Statistics for individual network devices are displayed.

Network Activity Table

The following illustrates the Network Activity Table:

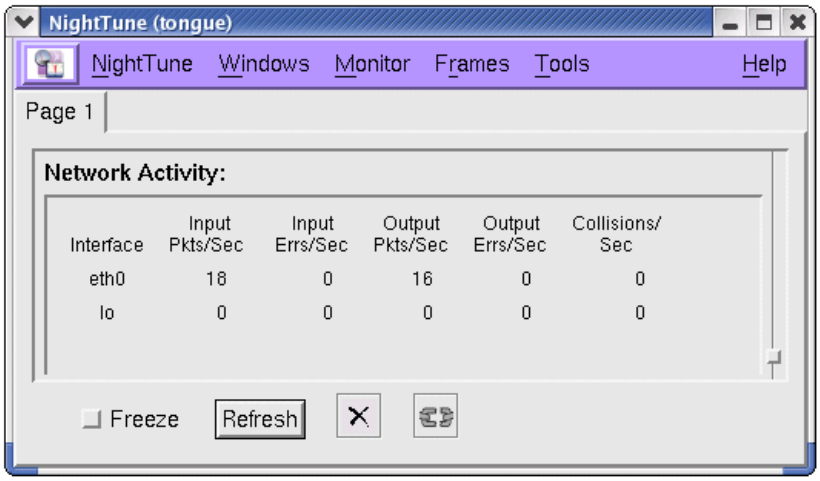

Figure 3-28. Network Activity Table

The information displayed in this area includes:

Interface

The name of the network interface is displayed in this column.

Input Rate

The number of packets transferred per second on behalf of input operations is displayed in this column.

Input Error Rate

The number of errors that occurred per second during input operations is displayed in this column.

Output Rate

The number of packets transferred per second on behalf of output operations is display in this column.

Output Error Rate

The number of errors that occurred per second during output operations is displayed in this column.

Collision Rate

The number of network collisions that occurred per second is displayed in this column.

The network activity data is automatically refreshed at a cyclic rate. The refresh interval can be adjusted using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Network Activity Line Graphs

The Network Activity Line Graphs provide individual line graphs detailing metrics related to network operations.

The following illustrates the Network Activity Line Graphs:

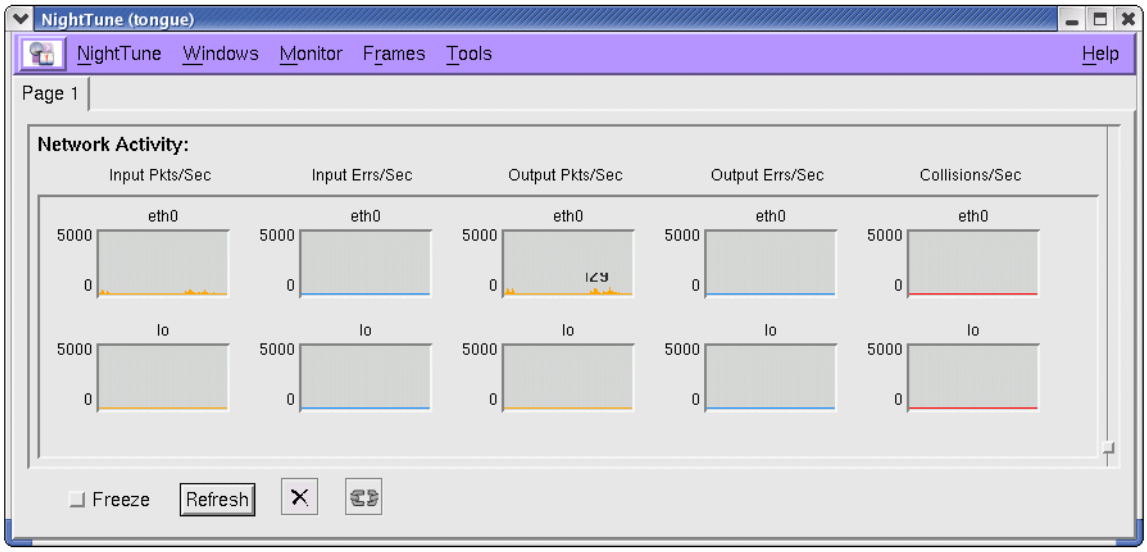

Figure 3-29. Network Activity Line Graphs

Individual line graphs are shown for each network device for each metric. The value of the metric is displayed vertically in each graph. The metrics include: Input Rate, Input Error Rate, Output Rate, Output Error Rate, and Collision Rate. See "Network Activity Table" on page 3-46 for definitions of these metrics.

The refresh interval can be changed using the Preferences dialog launched from the Preferences... option of the NightTune menu.

Network activity graphs are automatically scaled by NightTune based on peak network activity as monitoring continues. You can cause NightTune to rescale the graphs based on current network activity using the Rescale displays option from the Network Activity pop-up menu.

Network Activity Pop-up Menu

While positioned in the Network Activity panel, right-clicking launches the Network Activity pop-up menu.

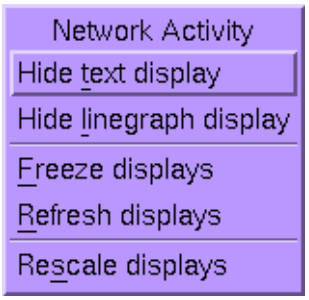

Figure 3-30. Network Activity pop-up menu

The following paragraphs describe the menu items in detail:

Show/Hide text display

This menu item toggles the visibility of the Network Activity Table within the panel.

Show/Hide linegraph display

This menu item toggles the visibility of the Network Activity Line Graphs within the panel.

Freeze/Unfreeze displays

This menu item toggles the Freeze setting for the Network Activity panel. When frozen, data values are not refreshed automatically. This menu item overrides the Freeze setting for the window, but only applies to displays within the Network Activity panel.

Refresh displays

This menu item causes all data in all displays within the Network Activity panel to be refreshed once, regardess of the Freeze setting.

Rescale displays

This menu item causes NightTune to rescale all graphs within the Network Activity panel based on current network activity.

4
Guide to Operations

This chapter guides you in operating NightTune to execute the following specific tasks:

- Monitoring User Processes
- Changing User Process Scheduling Attributes
- Shielding a CPU
- Changing the CPU Affinity of an Interrupt

Monitoring User Processes

Using the Monitor menu, select the Process Monitor and Process List panels from the menu.

Two panels will appear in the window as illustrated below:

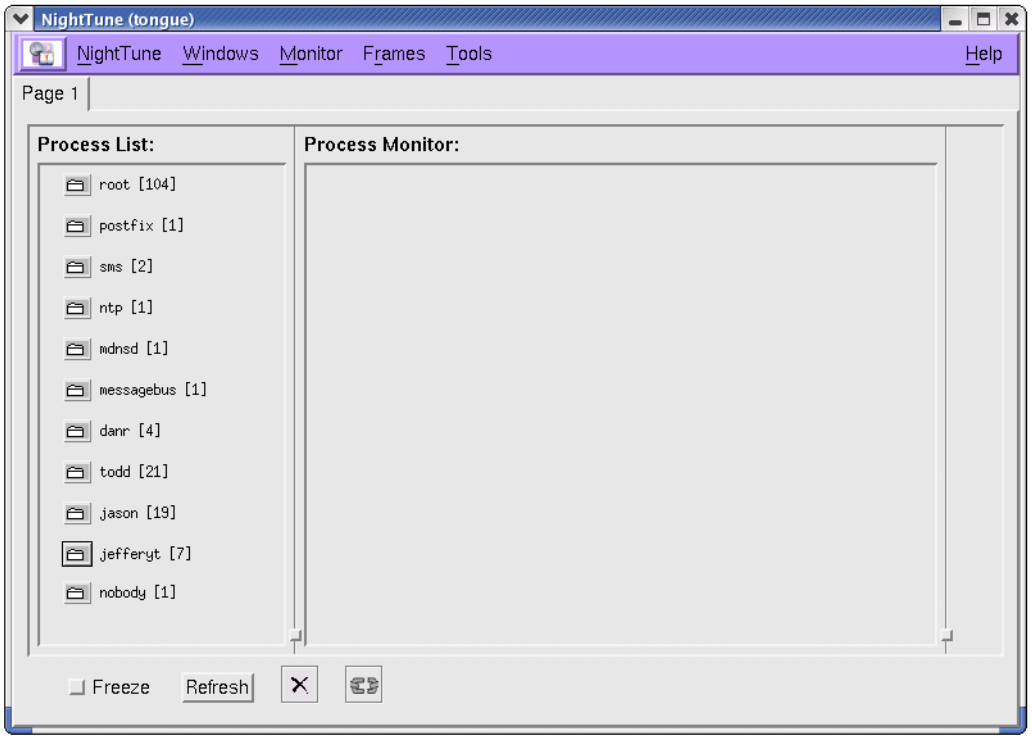

Figure 4-1. Monitoring User Processes

Selecting the User Process

The Process List panel is populated with User Frames which describe the number of processes associated with each user.

Click on the folder associated with the user name of the process you wish to monitor.

The list of processes is expanded beneath the folder as shown below:

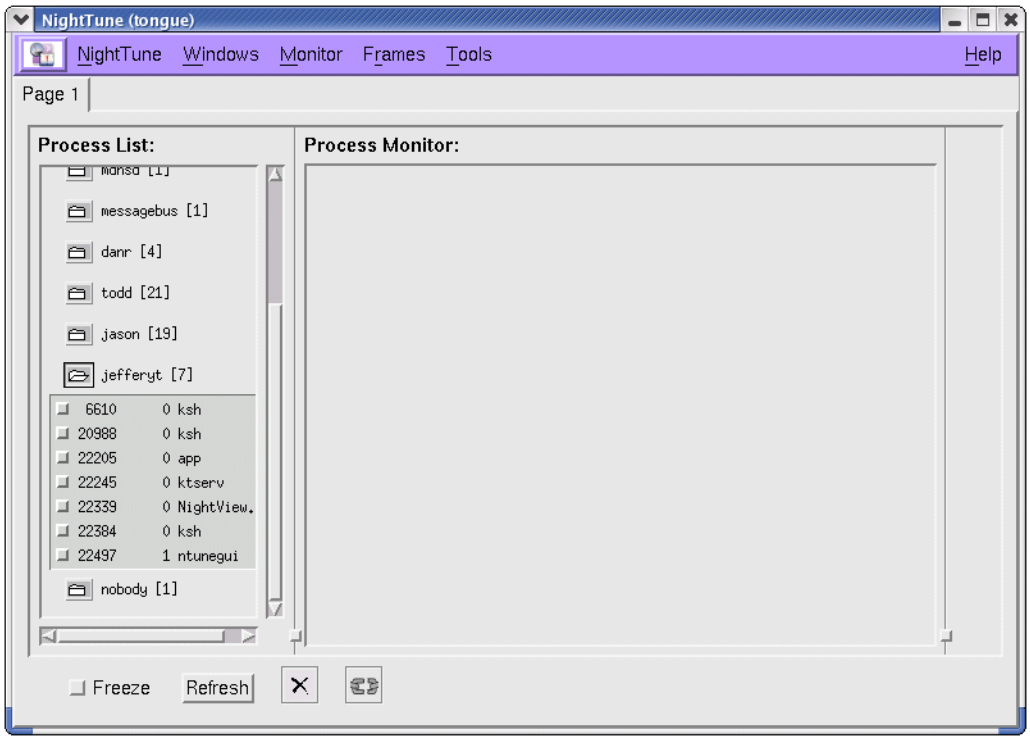

Figure 4-2. Selecting the User Process

Choose the processes you wish to monitor by selecting the checkboxes on the rows that describe the processes of interest.

This causes a User Frame to be opened in the Process Monitor panel; process attributes are displayed and updated cyclically, as shown below:

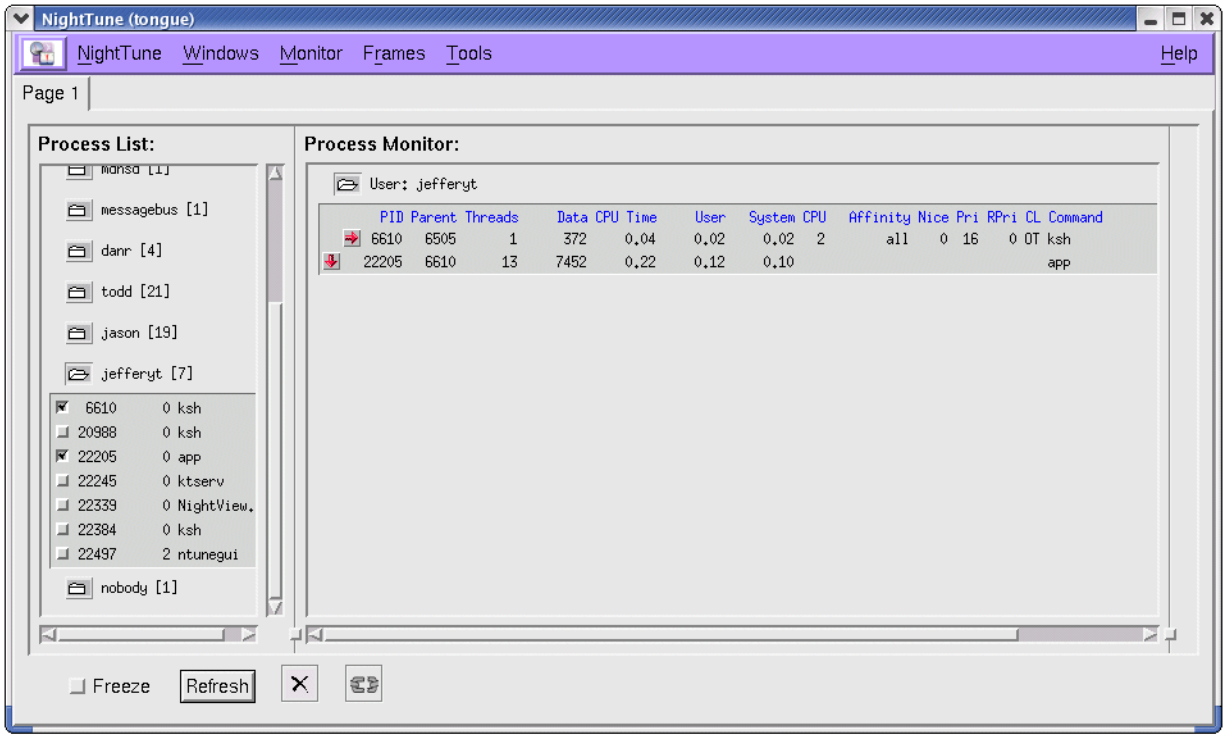

Figure 4-3. Monitoring Process Attributes

If a process is multi-threaded, a push-button with a vertical red arrow appears to the left of the process line. Clicking on the push-button causes the list of individual threads to be expanded, as shown below:

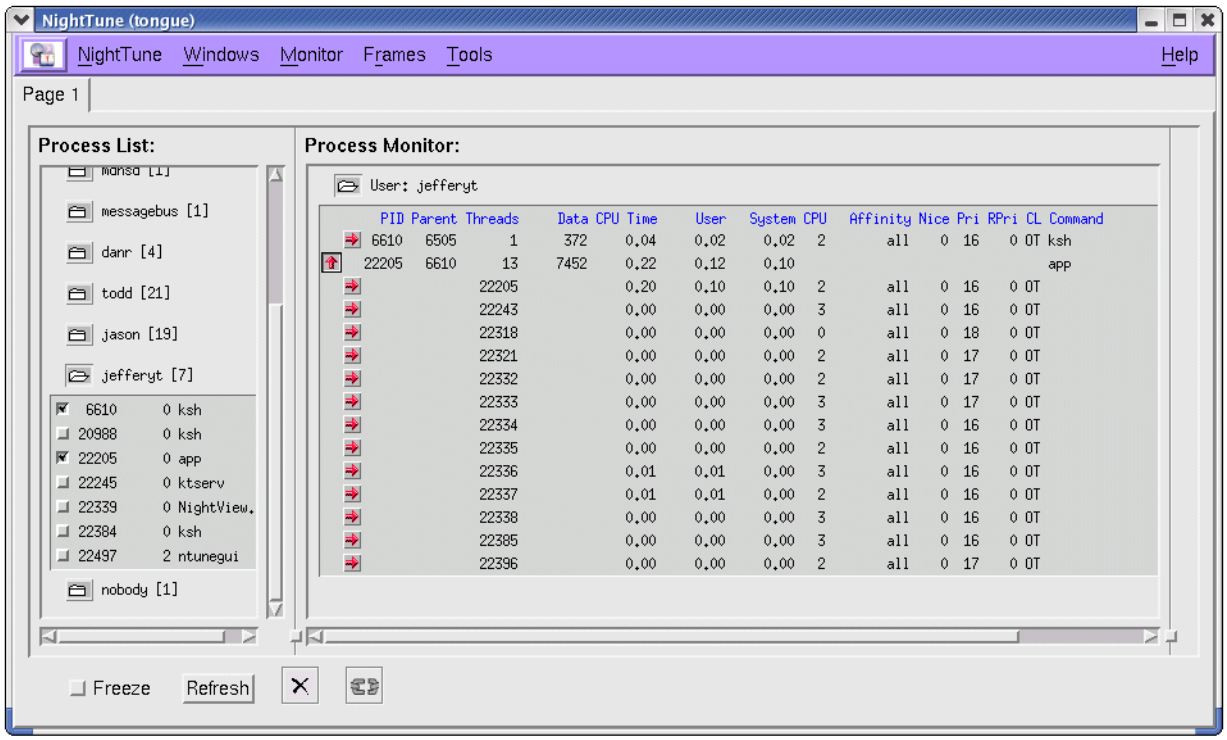

Figure 4-4. Monitoring Multi-threaded Processes

Customizing the Process Information

Select the Display Fields item from the Frames menu to launch the Process Fields menu.

NOTE

It is important to note that the Process Fields menu, like the other menus, can be "torn off" from the menu bar to reside in its own window separate from NightTune. This is especially useful when making multiple changes so as to avoid selecting the menu after every choice.

Select or deselect process fields of interest to customize the display.

In the illustration below, the following process fields have been selected for display:

- Process ID
- Thread Information
- Virtual Memory Size
- Resident Memory Size
- CPU Time
- Most Recent CPU
- Affinity Mask
- Real-time Priority
- Scheduling Class
- Command

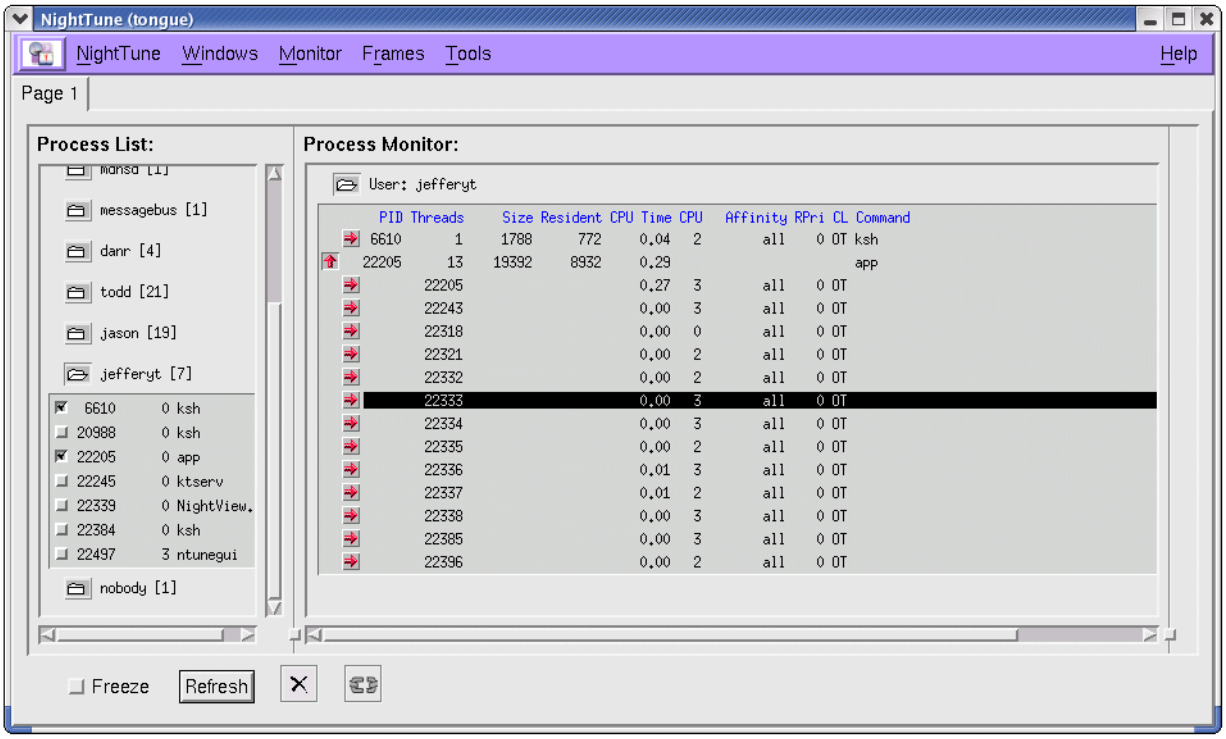

Figure 4-5. Customizing the Process Information

See “Process List Panel” on page 3-2 and “Process Monitor Panel” on page 3-6 for more information on process monitoring.

Changing User Process Scheduling Attributes

For each line that describes a thread (or single-threaded process), there is a push-button with a red arrow pointing to the right.

Click the push-button to launch the Process Scheduler dialog, as shown below:

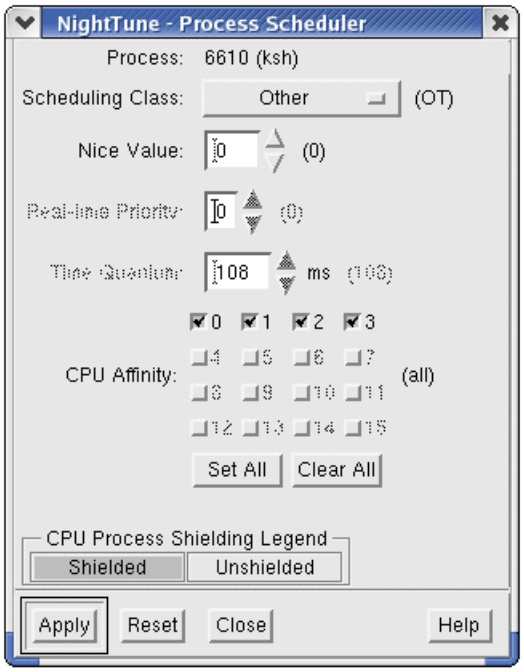

Figure 4-6. Process Scheduler dialog

The current scheduling attributes are shown in the dialog.

Change the Scheduling Class to Round Robin, the Real-time Priority to 20, and clear all CPU Affinity boxes except for CPU 0, which should be selected.

Click on Apply to apply the changes to the process; the Process Monitor panel display changes on the next refresh and indicates the new scheduling attributes, as shown below:

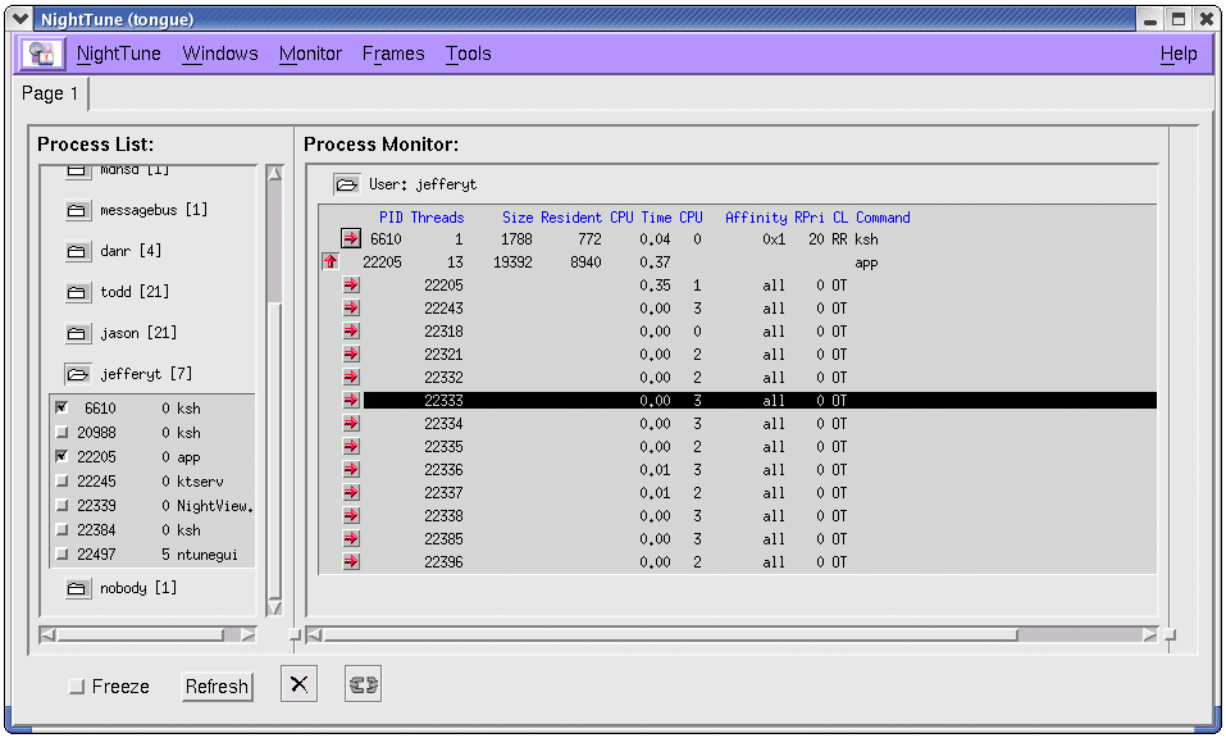

Figure 4-7. Changing User Process Scheduling Attributes

Using Drag and Drop to Change Process CPU Affinity

NightTune allows you to use drag and drop actions to change the CPU affinity of a process, group of processes, or individual threads.

Add the CPU Status panel to the tabbed page by selecting the CPU Status item from the Monitor menu. Remove the Process List panel from the tabbed page by clearing the checkbox in the Monitor menu.

The CPU Status panel is added to the tabbed page as shown below:

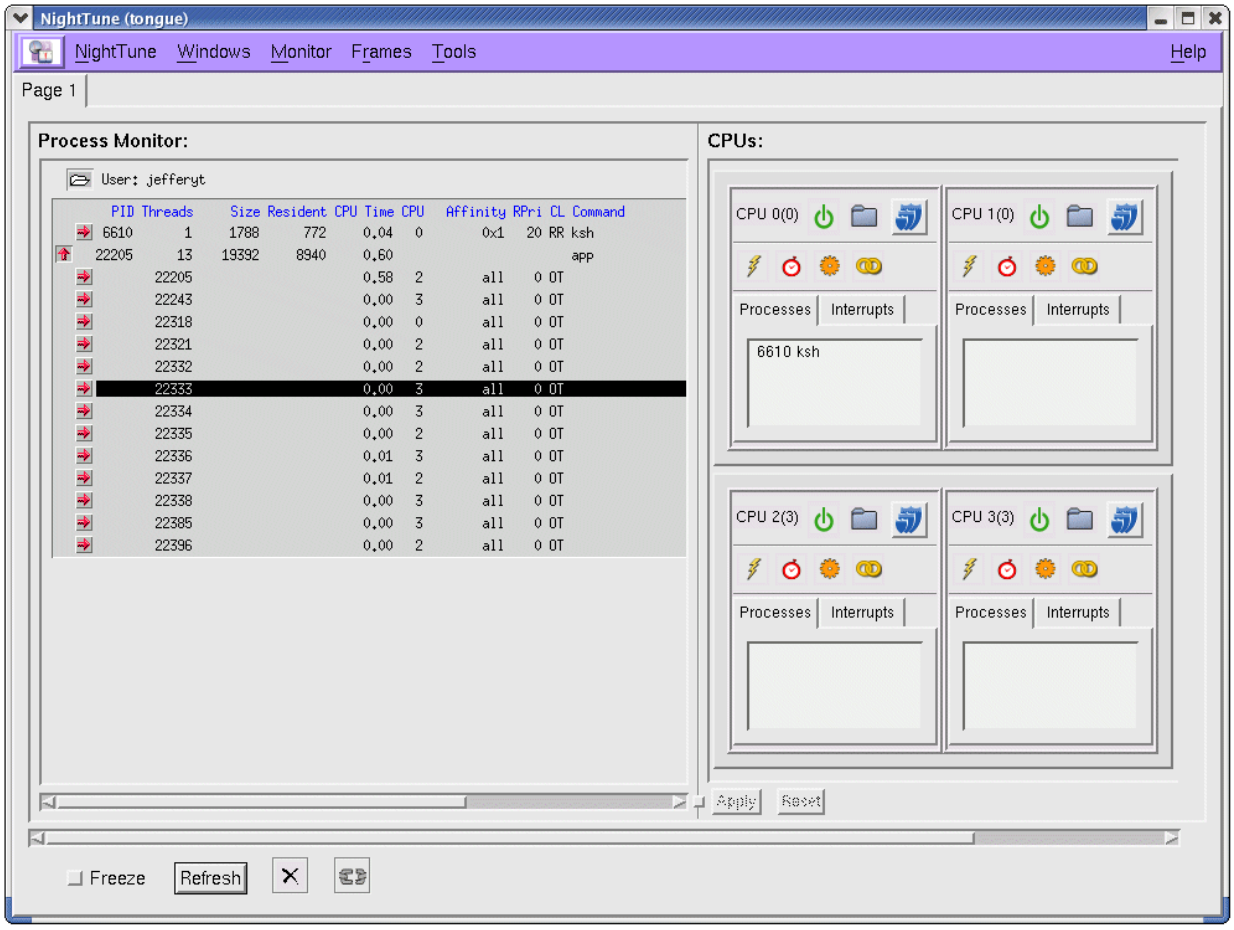

Figure 4-8. Viewing CPU Status

For each CPU displayed, a list of processes and interrupts that are bound to the CPU are also displayed.

A process or interrupt is considered bound to the CPU if the CPU affinity of the process or interrupt specifies a single CPU.

When you changed the CPU affinity of your process in the section above to CPU 0, it became bound to CPU 0. The process list in the CPU box for CPU 0 now shows the process.

To change the CPU of that process again, press and hold the middle mouse button over the process description in the process list in CPU box for CPU 0. Drag the pointer to CPU 1 and release the mouse button.

The new CPU affinity is reflected in the Process Monitor panel and in the bound process list for the CPU boxes as shown below:

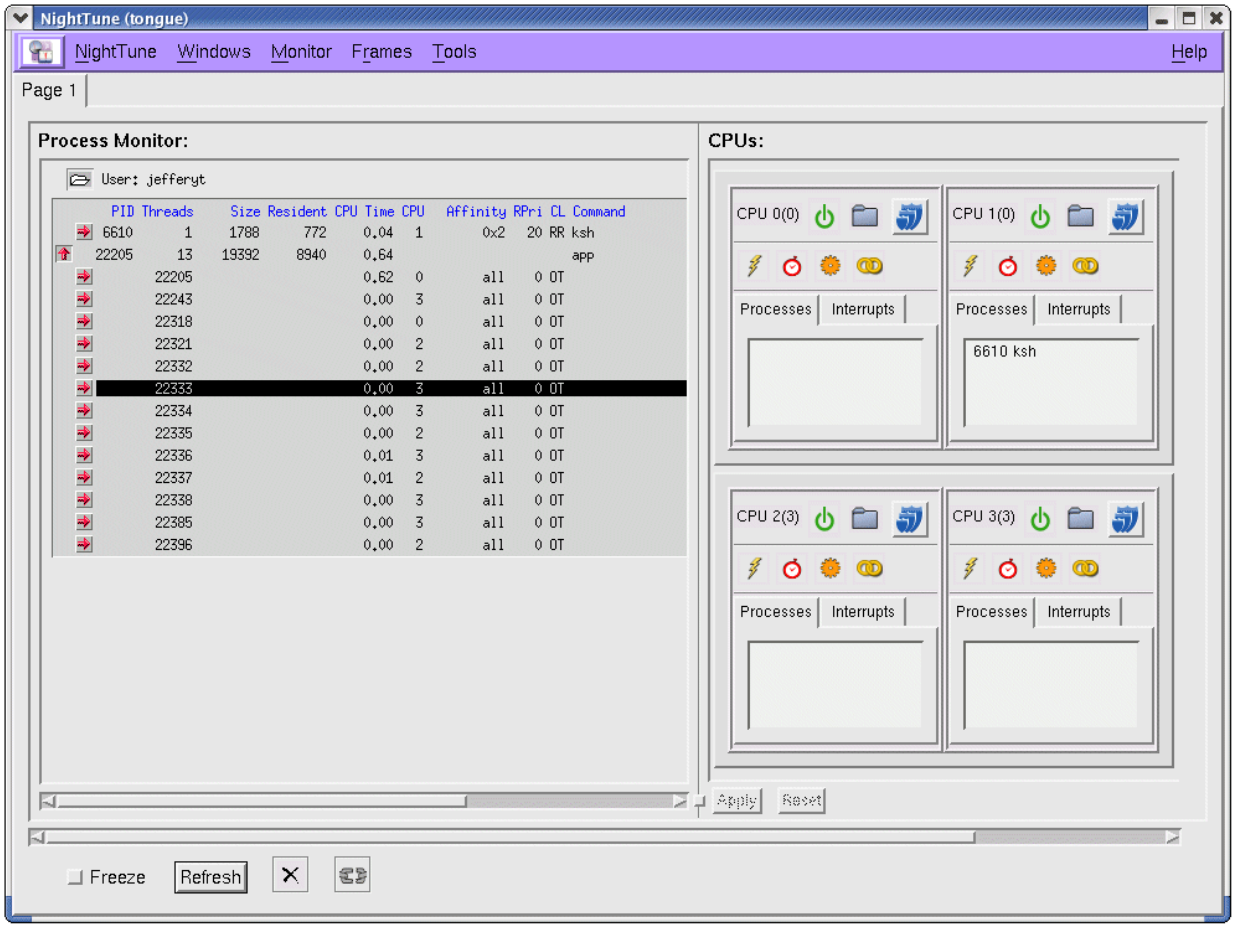

Figure 4-9. Using Drag and Drop to Change Process CPU Affinity

Similarly, you can drag a process or group of processes from the Process Monitor panel and drop them on a CPU box to change their CPU affinity.

See “Process Monitor Panel” on page 3-6 and “CPU Status Panel” on page 3-17 for more information on process CPU affinity.

Shielding a CPU

This section describes activities associated with shielding a CPU.

Close the Process Monitor panel from the window by clearing its checkbox in the Monitor menu.

Ensure that the CPU Status panel is in the window; select it from the Monitor menu if it is not already present.

The CPU Status panel below indicates that no shielding is currently active and that a single process has been bound to CPU 1.

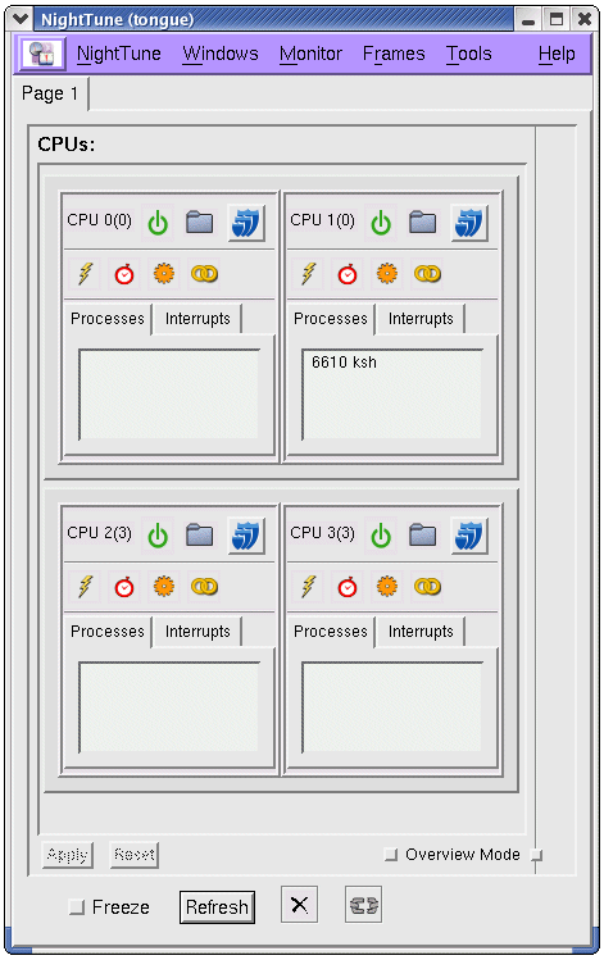

Figure 4-10. CPU Status Panel

The CPU Status panel illustration above describes a system with two physical CPUs which have hyper-threaded CPUs. CPU 0 and CPU 1 are the logical CPUs which share physical CPU 0; CPU 2 and CPU 3 are the logical CPUs which share physical CPU 3.

NOTE

Not all systems support hyper-threading. Systems with hyper-threading have two CPUs per row in the CPU Status panel. However, the sibling CPU is not always a logical CPU with a consecutive number. In this example, CPU 2 is CPU 0's hyper-threaded sibling.

To specify maximum shielding on CPU 0, click the maximum shield icon. 

The CPU Status panel display changes to indicate that CPU 0 is to be shielded from interrupts, processes, the local timer, and hyper-threading. To achieve that, CPU 1 will be marked Down. If your system does not support hyper-threading, CPU 1 will remain unchanged.

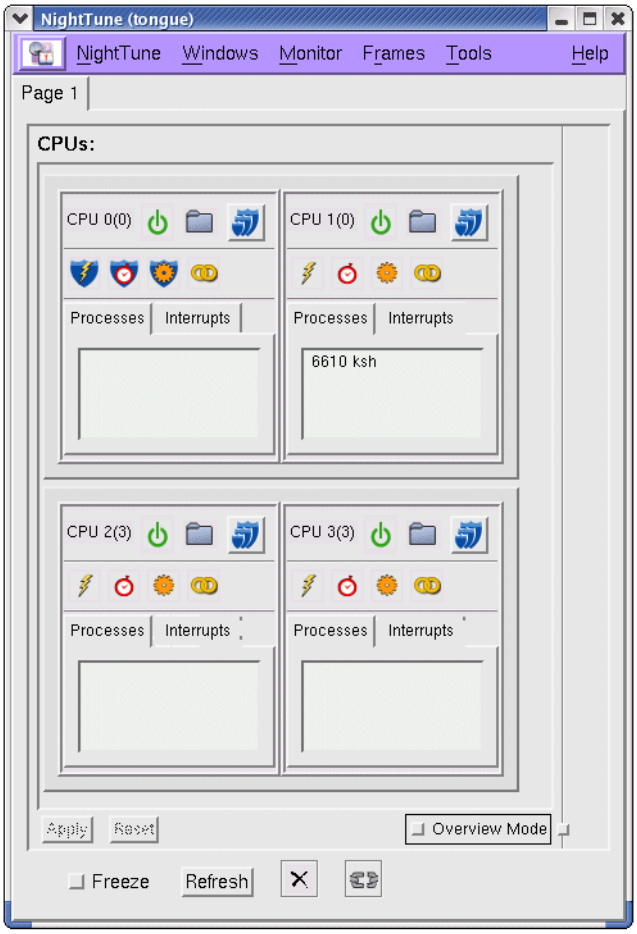

Figure 4-11. Shielding a CPU

Click on Apply to apply the shielding changes.

In the illustration above, there was a process bound to CPU 1. This prevents CPU 1 from being marked down. A diagnostic dialog similar to the following will appear if this situation exists:

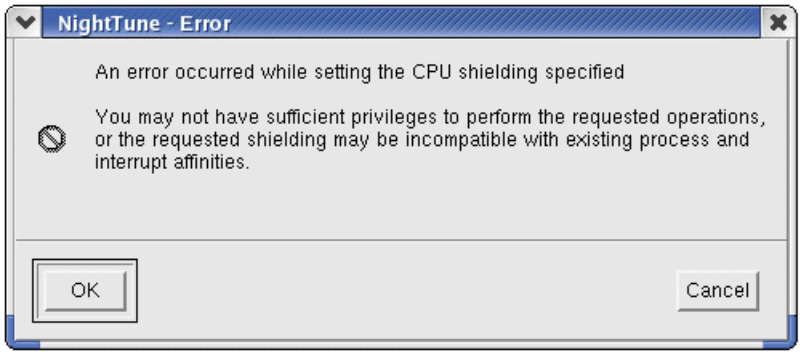

Figure 4-12. Error Shielding CPU

The process that is bound to CPU 1 must first be moved to another CPU before it can be marked Down.

Drag the process in the Processes list in CPU 1 using the middle mouse button from CPU 1 to CPU 2 or 3. Click the Max Shield icon on CPU 0.

Now click the shield icon the Apply button again and the CPU Status panel will reflect the changes as shown below:

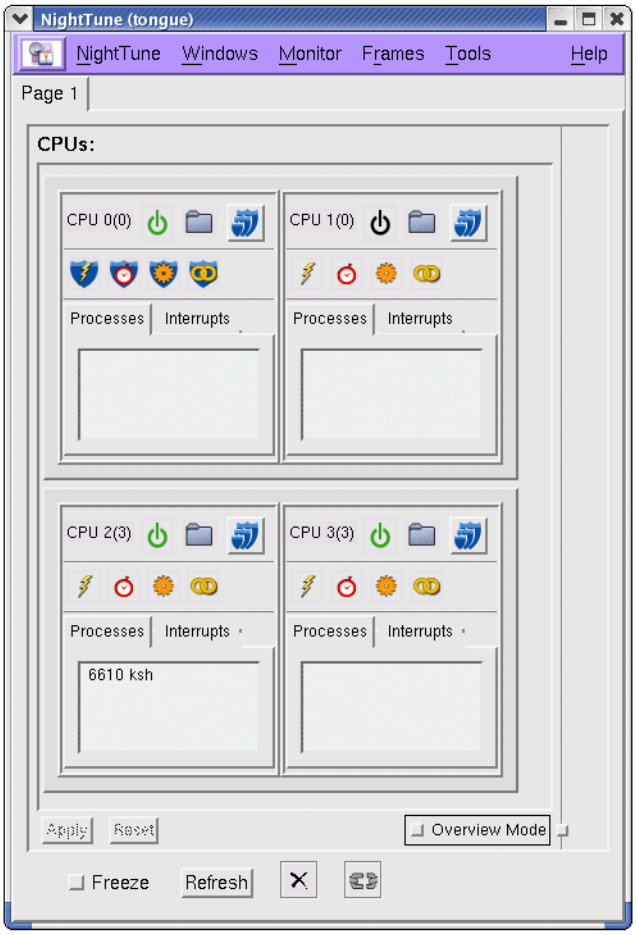

Figure 4-13. Maximum Shielding of a CPU

Interrupts may also be bound to specific CPUs. The Interrupts tab in the CPU box displays the list of interrupts that are bound. Bound interrupts prevent downing a CPU, just as bound processes do. Interrupts may be dragged from the list to other CPU boxes in the same manner as with processes.

See “CPU Status Panel” on page 3-17 for more information on CPU shielding.

Changing the CPU Affinity of an Interrupt

Add the Interrupt Activity panel to the window using the Monitor menu. Close all other panels if they exist by clearing their checkboxes in the Monitor menu.

While positioned in the Interrupt Activity panel, hide the bar graph and line graph panes by selecting the Hide bargraph display and Hide linegraph display items from the Interrupt Activity Pop-up Menu. The menu is launched by right-clicking while positioned in the Interrupt Activity panel.

The NightTune window will display the Interrupt Table and CPU Status information as shown below:

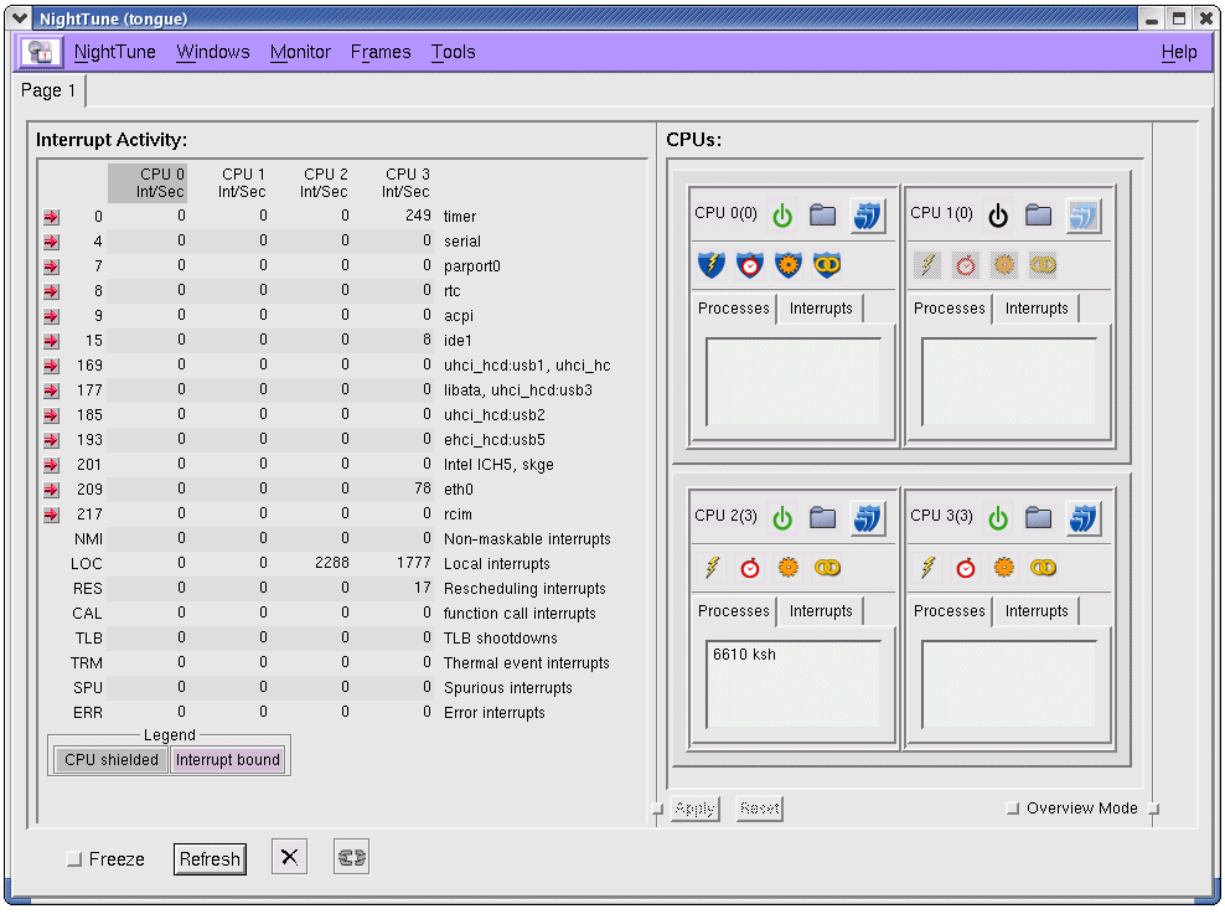

Figure 4-14. Monitoring Interrupt Activity

Click on the red-arrow push-button associated with the RCIM interrupt to launch the Interrupt Affinity dialog, as shown below:

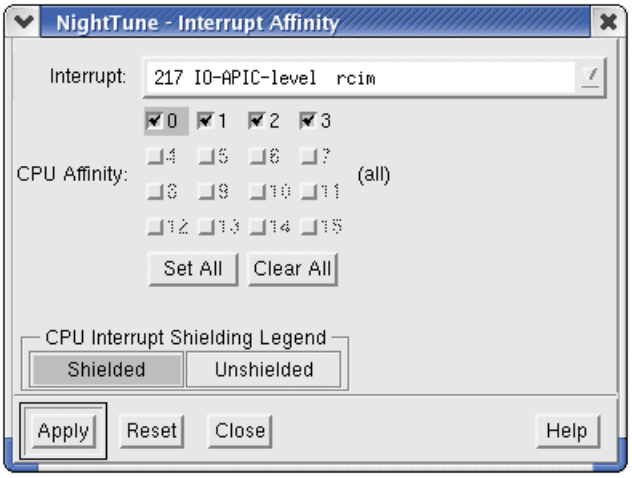

Figure 4-15. Interrupt Affinity dialog

Clear all the CPU affinity settings by clicking on Clear All. Then select CPU 0 by clicking on its checkbox. Apply the change by clicking on the Apply button.

The CPU affinity change is reflected in the Interrupt Activity panel. A purple background will now surround the cell for the RCIM interrupt on CPU 0, indicating that its affinity mask has selected CPU 0 but not all other CPUs.

Similarly, the RCIM interrupt will be added to the list of Interrupts in the CPU Status panel for CPU 0. If the list of interrupts is not currently being displayed, click on the Interrupts tab in the CPU box for CPU 0.

The illustration below reflects the affinity change for the RCIM interrupt:

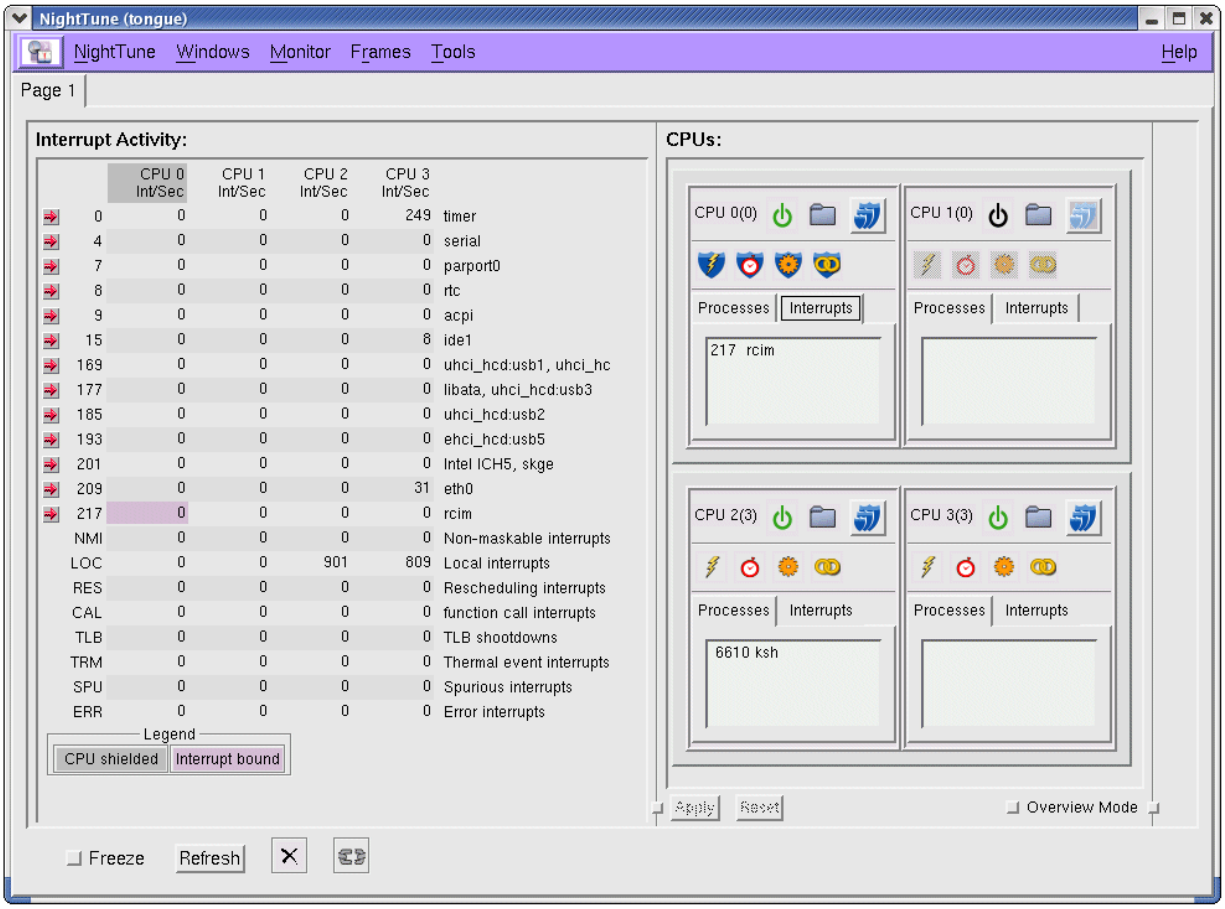

Figure 4-16. Changing the CPU Affinity of an Interrupt

Using Drag and Drop to Change Interrupt CPU Affinity

NightTune allows you to use drag and drop actions to change the affinity of an interrupt.

In the CPU box for CPU 0, middle click the line describing the RCIM interrupt (see section above) in the Interrupts list. Drag the pointer to another CPU and release the mouse button.

The Interrupt Activity Table and the Interrupts list in the destination CPU box reflect the change, as shown below:

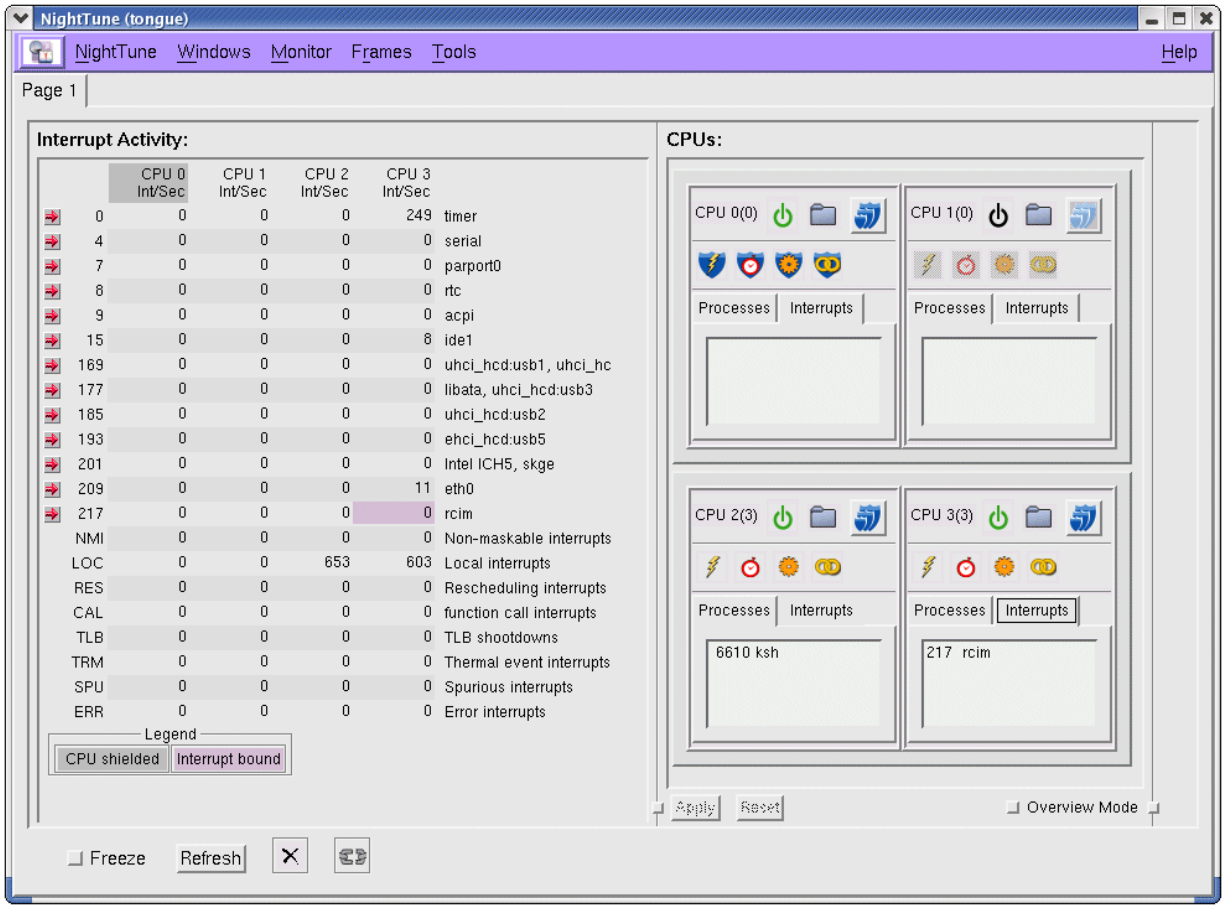

Figure 4-17. Using Drag and Drop to Change Interrupt CPU Affinity

See “Interrupt Activity Panel” on page 3-24 for more information on interrupt CPU affinity.

A
NightStar Licensing

NightStar RT uses the NightStar License Manager (NSLM) to control access to the NightStar RT tools.

License installation requires a licence key provided by Concurrent.The NightStar RT tools request a licence (see "License Requests" on page A-2) from a license server (see "License Server" on page A-2).

Two license modes are available, fixed and floating, depending on which product option you purchased. Fixed licenses can only be served to NightStar RT users from the local system. Floating licenses may be served to any NightStar RT user on any system on a network.

Tools are licensed per system, per concurrent user. A single license is shared among any or all of the NightStar RT tools for a particular user on a particular system. The intent is to allow *n* developers to fully utilize all the tools at the same time while only requiring *n* licenses. When operating the tools in remote mode, where a tool is launched on a local system but is interacting with a remote system, licenses are required only from the host system.

You can obtain a license report which lists all licenses installed on the local system, current usage, and expiration date for demo licenses (see "License Reports" on page A-3).

The default configuration includes a strict firewall which interferes with floating licenses. See "Firewall Configuration for Floating Licenses" on page A-3 for information on handling such configurations.

See "License Support" on page A-4 for information on contacting Concurrent for additional assistance with licensing issues.

License Keys

Licenses are granted to specific systems to be served to either local or remote clients, depending on the license model, fixed or floating.

License installation requires a license key provided by Concurrent. To obtain a license key, you must provide your system identification code. The system identification code is generated by the **`nslm_admin`** utility:

```
nslm_admin --code
```

System identification codes are dependent on system configurations. Reinstalling Linux on a system or replacing network devices may require you to obtain new license keys.

To obtain a license key, use the following URL:

http://www.ccur.com/NightStarRTKeys

Provide the requested information, including the system identification code. Your license key will be immediately emailed to you.

Install the license key using the following command:

nslm_admin --install=*xxxx-xxxx-xxxx-xxxx-xxxx*

where *xxxx-xxxx-xxxx-xxxx-xxxx* is the key included in the license acknowledgment email.

License Requests

By default, the NightStar RT tools request a license from the local system. If no licenses are available, they broadcast a license request on the local subnet associated with the system's hostname.

You can control the license requests for an entire system using the **/etc/nslm.config** configuration file.

By default, the **/etc/nslm.config** file contains a line similar to the following:

```
:server @default
```

The argument @default may be changed to a colon-separated list of system names, system IP addresses, or broadcast IP addresses. Licenses will be requested from each of the entities found in the list, until a license is granted or all entries in the list are exhausted.

For example, the following setting prevents broadcast requests for licenses, by only specifying the local system:

```
:server localhost
```

The following setting requests a license from **server1**, then **server2**, and then a broadcast request if those fail to serve a license:

```
:server server1:server2:192.168.1.0
```

Similarly, you can control the license requests for individual invocations of the tools using the **NSLM_SERVER** environment variable. If set, it must contain a colon-separated list of system names, system IP addresses, or broadcast IP addresses as described above. Use of the **NSLM_SERVER** environment variable takes precedence over settings defined in **/etc/nslm.config**.

License Server

The NSLM license server is automatically installed and configured to run when you install NightStar RT.

The **nslm** service is automatically activated for run levels 2, 3, 4, and 5. You can check on these settings by issuing the following command:

```
/sbin/chkconfig --list nslm
```

In rare instances, you may need to restart the license server via the following command:

```
/sbin/service nslm restart
```

See **nslm(1)** for more information.

License Reports

A license report can be obtained using the **nslm_admin** utility.

```
nslm_admin --list
```

lists all licenses installed on the local system, current usage, and expiration date (for demo licenses). Use of the **--verbose** option also lists individual clients to which licenses are currently granted.

Adding the **--broadcast** option will list this information for all servers that respond to a broadcast request on the local subnet associated with the system's hostname.

See **nslm_admin(1)** for more options and information.

Firewall Configuration for Floating Licenses

RedHawk does not support a firewall configuration by default, because iptables support is disabled. However, it is possible to build a custom kernel with iptables support enabled. If that is done, and floating licenses are used, the iptables firewall rules must be configured to allow the license requests and responses to pass.

If the system with iptables support and firewall rules is serving licenses, then the firewall rules must be arranged to allow license requests on UDP port 25517 and TCP port 25517 from any systems that will make license requests. For example, in a simple firewall, rules like the following, inserted before any DROP or REJECT rules, might work:

```
iptables -A INPUT -p udp -m udp -s subnet/mask --dport 25517 -j ACCEPT
iptables -A INPUT -p tcp -m tcp -s subnet/mask --dport 25517 -j ACCEPT
```

If the system with iptables support and firewall rules is running NightStar RT tools and receiving floating licenses, then the firewall rules must be arranged to allow license responses on UDP port 25517 from any system serving licenses. For example, in a simple firewall, rules like the following, inserted before any DROP or REJECT rules, might work:

```
iptables -A INPUT -p udp -m udp -s subnet/mask --sport 25517 -j ACCEPT
```

License Support

For additional aid with licensing issues, contact the Concurrent Software Support Center at our toll free number 1-800-245-6453. For calls outside the continental United States, the number is 1-954-283-1822. The Software Support Center operates Monday through Friday from 8 a.m. to 5 p.m., Eastern Standard Time.

You may also submit a request for assistance at any time by using the Concurrent Computer Corporation web site at **http://www.ccur.com/isd_support_contact.asp** or by sending an email to **support@ccur.com**.

B
Kernel Dependencies

Concurrent's RedHawk kernel provides features and performance gains that are critical for the full operation of the NightStar RT tools.

The NightStar RT tools can operate in a host-only mode on Red Hat systems without Concurrent's RedHawk kernel, cross-targeting to RedHawk systems. Additionally, the NightStar RT tools can function on Red Hat systems without the RedHawk kernel, but will lack the numerous advantages afforded by running with it.

The following sections describe the additional functionality and capabilities of the NightStar RT tools when running Concurrent's RedHawk kernel.

Advantages for NightView

The following advantages are afforded NightView when Concurrent's RedHawk kernel is running:

- Application speed conditions

  Provides "execution-speed" patches, conditions, and ignore counts.

- Hot operations

  Users of NightView gain the ability to read and write to a particular process without having to stop it. Thus, all eventpoints can be applied and modified during application program execution without stopping the process. User variables also can be read and modified without stopping the process.

- Signal handling

  Allows NightView to pass signals directly to a particular process, avoiding context switching.

NOTE

NightView may not operate at all on older versions of Red Hat without the RedHawk kernel.

Advantages for NightTrace

The following advantage is afforded NightTrace when Concurrent's RedHawk tracing kernel is running:

- Kernel tracing

  Users of NightTrace gain the ability to obtain kernel trace data and combine that with user trace data. Kernel tracing is an incredibly powerful feature that not only provides insight into the operating system kernel but also provides useful information relating to the execution of user applications.

The RedHawk kernel is provided in three flavors:

- Tracing
- Debug
- Plain

The Tracing and Debug flavors provide the features required for NightTrace kernel tracing. These kernels can be selected at boot-time from the boot-loader menu.

Advantages for NightProbe

The following advantages are afforded NightProbe when Concurrent's RedHawk kernel is running:

- Minimal intrusion

  Allows NightProbe to read and write variables without stopping the process for each sample or write operation.

- Sampling performance

  Allows NightProbe to use direct memory fetches for data sampling (as opposed to programmed I/O) which is important for high-rate data acquisition.

- Concurrent debugging/probing

  Allows NightProbe to probe programs already under the control of a debugger or another NightProbe session.

- PCI Device probing

  Allows NightProbe to probe PCI device memory via the Base Address Register (BAR) file system.

Advantages for NightTune

The following advantage is afforded NightTune when Concurrent's RedHawk kernel is running:

- Context switch rate

  Allows NightTune user to display the context switch counts per CPU instead of for the overall system.

- CPU shielding

  Individual CPUs can be shielded from interrupts and processes allowing CPUs to be dedicated solely to specific interrupts and processes that are bound to the CPU.

- CPU sibling interference

  Individual CPUs can be marked down to avoid interfering with hyperthreaded sibling CPUs and dual-core sibling CPUs. Hyperthreaded CPUs share all the resources of their sibling CPU. Dual-core CPUs share the CPU cache and a path to memory with their sibling CPU.

Advantages for NightSim

The following advantage is afforded NightSim when Concurrent's RedHawk kernel is running:

- Scheduling target

  Allows NightSim to schedule processes on the system via Concurrent's Frequency-Based Scheduler.

Index

L

M

N

O

P