



NightSim RT User's Guide

Version 4.5

(RedHawk™ Linux®)

Copyright 2010,2018 by Concurrent Real-Time, Inc.. All rights reserved. This publication or any part thereof is intended for use with Concurrent products by Concurrent personnel, customers, and end-users. It may not be reproduced in any form without the written permission of the publisher.

The information contained in this document is believed to be correct at the time of publication. It is subject to change without notice. Concurrent Real-Time, Inc. 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 Real-Time, Inc., 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.

Concurrent Real-Time, Inc. and its logo are registered trademarks of Concurrent Real-Time, Inc. All other Concurrent Real-Time, Inc. product names are trademarks of Concurrent Real-Time, Inc. while all other product names are trademarks or registered trademarks of their respective owners.

Linux[®] is used pursuant to a sublicense from the Linux Mark Institute.

NightStar’s integrated help system is based on Assistant, a Qt[®] utility. Qt is a registered trademark of Digia Plc and/or its subsidiaries.

NVIDIA[®] CUDA[™] is a trademark of NVIDIA Corporation.

Scope of Manual

This guide is designed to assist you in getting started with the use of NightSim™. NightSim is a real-time NightStar™ RT tool that provides a graphical user interface to the frequency-based scheduler and performance monitor services.

Structure of Manual

This manual consists of five chapters, four appendixes, a glossary, and an index. A brief description of the chapters and appendixes is presented as follows.

- Chapter 1 introduces you to the concepts and components of NightSim, a real-time tool that is part of the NightStar™ development environment.
- Chapter 2 provides an overview of the primary factors that need to be taken into account prior to running NightSim.
- Chapter 3 explains the procedures for beginning and ending a NightSim session and explains how to use the Menu bar and toolbar.
- Chapter 4 describes the frequency-based scheduler and introduces the components of NightSim's Scheduler and Processes areas.
- Chapter 5 describes the performance monitor and introduces the components of NightSim's Monitor area.
- Appendix A discusses the NightStar License Manager (NSLM) and how to obtain and install licenses. It also discusses approaches for dealing with a firewall either on the system acting as the license server or on a system hosting the NightStar RT tools.

Appendix B describes dependencies of certain features on the RedHawk kernel. The Glossary contains definitions of technical terms that are important to understanding the concepts presented in this book.

The Index contains an alphabetical reference to key terms and concepts and numbers of pages where they occur in the text.

Syntax Notation

The following notation is used throughout this manual:

italic

Books, reference cards, and items that the user must specify appear in *italic* type. Special terms may also appear in *italics*.

list bold

User input appears in **list bold** type and must be entered exactly as shown. Names of directories, files, commands, options and system manual 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 appear in list type.

[]

Brackets enclose command options and arguments that are optional. You do not type the brackets if you choose to specify such options or arguments

Referenced Publications

The following publications are referenced in this document:

0890514	<i>NightBench User's Guide</i>
0898007	<i>Real-Time Clock and Interrupt Module (RCIM) User's Guide</i>
0898004	<i>RedHawk Linux™ User's Guide</i>
0898008	<i>NightStar RT Installation Guide</i>
0898009	<i>NightStar RT Tutorial</i>
0898395	<i>NightView RT User's Guide</i>
0898398	<i>NightTrace RT User's Guide</i>
0898465	<i>NightProbe RT User's Guide</i>
0898515	<i>NightTune RT User's Guide</i>
0898537	<i>MAXAda for Linux Reference Manual</i>

Contents

1 Overview

Overview of NightSim	1-1
Overview of the Frequency-Based Scheduler	1-1
Overview of the Performance Monitor.	1-2
Overview of the Graphical User Interface.	1-2
Overview of RCIM Support	1-4

2 Establishing the NightSim Environment

Configuration Requirements	2-1
User Requirements	2-1
Remote System Access.	2-1
Capabilities.	2-2

3 Getting Started

Invoking NightSim	3-1
The Menu Bar	3-3
File Menu	3-3
Target Menu	3-6
View Menu	3-7
Scheduler Menu.	3-8
Monitor Menu.	3-8
Tools Menu.	3-8
Help Menu	3-10
The Toolbar	3-12
Using Accelerators.	3-13
Preferences Dialog.	3-14
General Preference Page.	3-16
Fonts Preference Page	3-17
NightStar Global Fonts Dialog	3-21
Exiting NightSim.	3-23

4 Using the Scheduler Area

Understanding the Frequency-Based Scheduler	4-1
How Is Scheduler Frequency Defined?	4-2
How Are Processes Scheduled?	4-2
Programming Requirements.	4-5
The Scheduler Menu	4-5
Using the Scheduler Control Area	4-9
Using the Scheduler Definition Area.	4-12
Using the Scheduler Interrupt Area	4-13
Using the Processes Area.	4-15

The Process Menu	4-16
Add Process	4-19
Edit Process	4-21
Add Process and Edit Process Dialog Pages	4-22
FBS Schedule	4-22
Runtime Properties	4-25
I/O and Debug	4-28
Pathname Interpretation.	4-30

5 Using the Monitor Area

Understanding the Performance Monitor.	5-1
What Values Are Monitored?	5-3
Monitoring Idle Time	5-6
How Is Idle Time Monitored?	5-6
Monitoring Unscheduled Processes	5-6
The Monitor Page.	5-8
The Monitor Menu	5-9
The Metrics Panel	5-11
The Percent of Period Used Panel	5-17
Logging	5-21
Logging Dialog	5-22
Logging Status Bar	5-24

Appendix A NightStar RT Licensing

License Keys	A-1
License Requests	A-2
License Server	A-2
License Reports	A-3
Firewall Configuration for Floating Licenses	A-3
License Support	A-4

Appendix B Kernel Dependencies

Advantages for NightView.	B-1
Advantages for NightTrace	B-1
Advantages for NightProbe	B-2
Advantages for NightTune.	B-3
Frequency Based Scheduler	B-3

Glossary

Index

Illustrations

Figure 1-1. The NightSim Window	1-3
Figure 3-1. File Menu	3-3

Figure 3-2. Target Menu	3-6
Figure 3-3. View Menu	3-7
Figure 3-4. Scheduler Menu	3-8
Figure 3-5. Monitor Menu	3-8
Figure 3-6. Tools Menu	3-9
Figure 3-1. Help Menu	3-10
Figure 3-2. NightSim Toolbar	3-12
Figure 3-3. General Preference Page	3-16
Figure 3-4. Fonts Preferences Page	3-19
Figure 3-5. NightStar Global Fonts Dialog	3-21
Figure 4-1. Scheduler Menu	4-5
Figure 4-2. Select Active Scheduler Dialog	4-6
Figure 4-3. Scheduler Control Area	4-9
Figure 4-4. Scheduler Definition Area	4-12
Figure 4-5. Scheduler Interrupt Area	4-13
Figure 4-6. Interrupt Source Dropdown Menu	4-14
Figure 4-7. Clock Period/Freq Dropdown	4-15
Figure 4-8. Processes Area	4-16
Figure 4-9. Process Menu	4-17
Figure 4-10. Process Columns to Display Dialog	4-19
Figure 4-11. Add Process Dialog	4-20
Figure 4-12. Add Process and Edit Process Dialogs - FBS Schedule Page	4-22
Figure 4-13. Add Process and Edit Process Dialog - Runtime Properties Page	4-26
Figure 4-14. Add Process and Edit Process Dialog - I/O and Debug Page	4-28
Figure 5-1. Monitor Page	5-8
Figure 5-2. Monitor Menu	5-9
Figure 5-3. Resolution Options	5-10
Figure 5-4. The Metrics Panel	5-11
Figure 5-5. The Metrics Panel Context Menu	5-12
Figure 5-6. Select Monitor Columns Dialog	5-14
Figure 5-7. Resolution Options	5-15
Figure 5-8. Sort Key Selection Dialog	5-16
Figure 5-9. Percent of Period Used Panel	5-17
Figure 5-10. Panel Context Menu	5-20
Figure 5-11. Logging Dialog	5-22
Figure 5-12. Logging Status Bar	5-24

Tables

Table 2-1. Recommended /etc/pam.d Configuration	2-3
Table 4-1. Process Scheduling	4-3
Table 4-2. Scheduler Operation	4-3

This chapter introduces you to the concepts and components of NightSim, a real-time tool that is part of the NightStar development environment. NightSim provides a graphical user interface to the major functions associated with the *frequency-based scheduler* and the *performance monitor*.

Overview of NightSim

NightSim allows you to perform the entire range of functions associated with frequency-based scheduling. You can perform the major functions of configuring a scheduler, setting up an interrupt source, scheduling programs, saving and restoring a scheduler configuration, running a simulation, and viewing scheduling data. The principal features of the *frequency-based scheduler* are described in the section that follows.

NightSim also allows you to perform the entire range of functions associated with the *performance monitor*. You can perform the major functions of selecting a scheduler, clearing existing performance monitor values, enabling and disabling performance monitoring, including and excluding interrupt time, and viewing performance monitor values. The principal features of the performance monitor are described in “Overview of the Performance Monitor” on page 1-2.

In addition, NightSim supports the use of interrupt sources, including Distributed Interrupts devices on the RCIM (Real-Time Clock and Interface Module) board.

NightSim is also integrated with other NightStar tools. NightSim may be invoked directly from the **TOOLS** menu of any NightStar RT tool, and NightSim provides access to every other NightStar RT tool in the same way. NightSim also provides the ability to schedule your programs under the control of the NightView™ Source-Level Debugger, making it easy to gain control of scheduled programs in order to set breakpoints or monitorpoints early on in the execution of the program, before it attaches to the frequency based scheduler.

Overview of the Frequency-Based Scheduler

The *frequency-based scheduler*, or FBS, is a task synchronization mechanism that allows you to run processes at specified frequencies. Frequencies can be based on high-resolution clocks, an external interrupt source, or completion of a cycle. The frequency-based scheduler provides a mechanism for initiating processes at the specified frequency. The processes are then scheduled via the standard priority-based scheduler.

When used in conjunction with the performance monitor, the FBS can be used to determine the best way of allocating processors to various tasks for a particular application. “Using the Scheduler Area” on page 4-1 fully describes the features and capabilities of the frequency-based scheduler.

Overview of the Performance Monitor

The performance monitor is a mechanism that allows you to monitor use of the CPU by processes that are scheduled on a frequency-based scheduler. The values obtained can help in determining whether processes need to be redistributed among processors for improved load balancing and processing efficiency. You can obtain performance monitor values by process and can specify whether time spent servicing interrupts should be included or excluded from the values. The performance monitor also makes it possible to monitor a processor's *idle time*. By examining the amount of idle time on each processor, you can identify the processors that have the lightest load and determine the additional amount of CPU time that can be used for scheduling real-time processes. “Using the Monitor Area” on page 5-1 fully describes the features and capabilities of the performance monitor.

Overview of the Graphical User Interface

NightSim's graphical user interface consists of a single window with tabbed pages that provide access to the key operations associated with frequency-based scheduling and performance monitoring. An overview of the menu selections and areas is provided in the sections that follow.

When NightSim is launched without arguments, the following window displays.

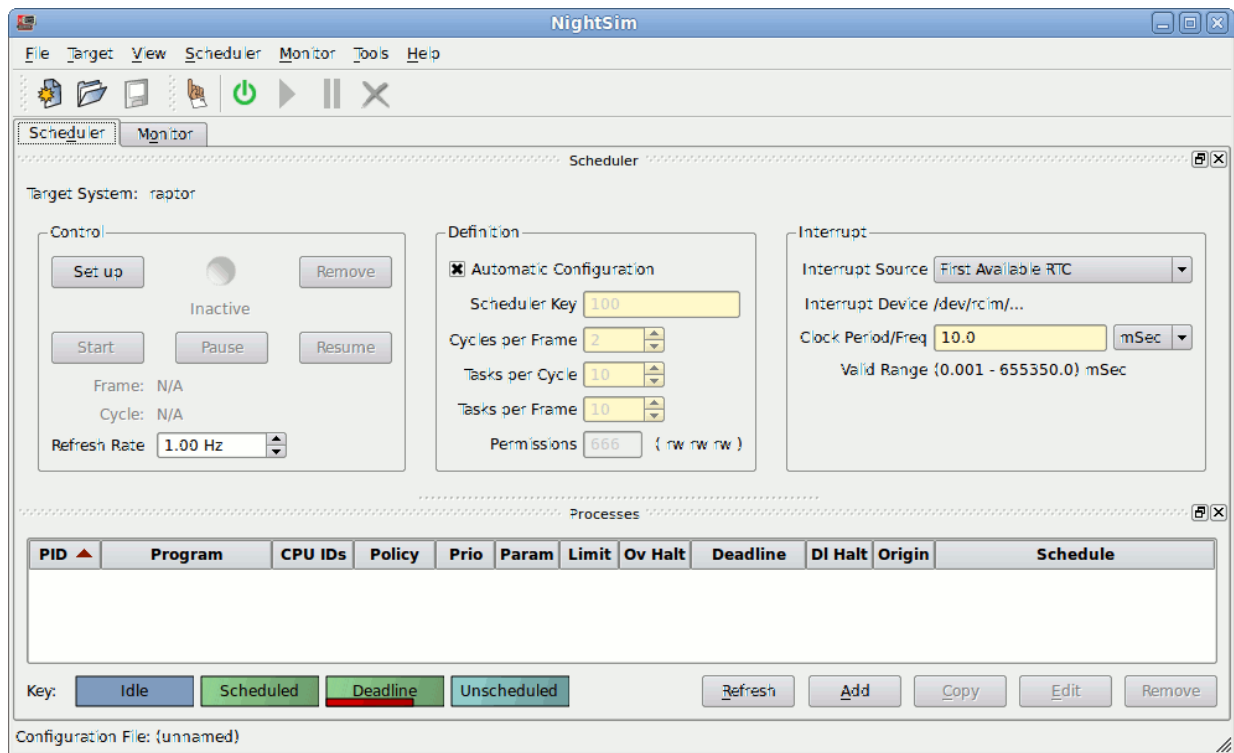


Figure 1-1. The NightSim Window

The main window presents two tabbed pages along with common menu items and toolbars for controlling NightSim.

The Scheduler page consists of the Scheduler and Processes panels.

The Scheduler panel consists of three areas:

- Control

This area allows you to create and remove a *frequency-based scheduler* and start, stop and resume a simulation. The status of the simulation run is also provided. Refer to the section “Using the Scheduler Control Area” on page 4-9 for a full description.

- Definition

This area allows you to configure a *frequency-based scheduler*, specify maximum values for the scheduler, and specify the permissions that are to be associated with it. Refer to the section “Using the Scheduler Definition Area” on page 4-12 for a full description.

- Interrupt

This area allows you to select the interrupt source that is to be used for the scheduler. Refer to the section “Using the Scheduler Interrupt Area” on page 4-13 for a full description.

The **Processes** panel allows you to select processes, add, edit, copy and remove processes from a scheduler and view data for the scheduled processes. Refer to the section “Using the Processes Area” on page 4-15 for a full description.

The **Monitor** page appears by clicking on the **Monitor** tab at the top of the screen or by selecting **Monitor** from the **View** menu or using the **Alt+O** keyboard shortcut. Performance Monitor data for the scheduled processes can be viewed in this area. Refer to “Using the Monitor Area” on page 5-1 for a full description.

Similarly, the **Scheduler** page can be shown by clicking on the **Scheduler** tab at the top of the screen or by selecting **Scheduler Page** from the **View** menu or using the **Alt+D** keyboard shortcut.

The toolbar at the top of the NightSim window contains icons for commonly used actions. The toolbar is described in the section “The Toolbar” on page 3-12.

Overview of RCIM Support

NightSim supports the Real-Time Clock and Interrupt Module (RCIM). An RCIM is a hardware module attached to a *target system* which contains the following interrupt-generating devices:

- real-time clocks (RTCs)
- edge triggered interrupts (ETIs)
- distributed interrupts (Distribs)
- programmable interrupt generators (PIGs)

The first three of these devices may be used as the interrupt source for an FBS; the latter PIG devices can be configured to cause distributed interrupt devices to fire.

RCIMs may be connected via RCIM cables such that interrupts from one RCIM may be distributed to other systems in the RCIM chain.

Each interrupt device on an RCIM may be configured as an *RCIM-distributed* or normal device. By default, each interrupt device is available for use on the attached target system. If the interrupt device is configured as an RCIM-distributed device, interrupts are also distributed between target systems using the RCIM cable.

NightSim provides the ability to create, control and monitor performance of individual schedulers and to control them from any host with network connectivity to the target.

For more information see the *Real-Time Clock and Interrupt Module (RCIM) User's Guide (0898007)*.

Establishing the NightSim Environment

This chapter provides an overview of the primary factors that need to be taken into account prior to running NightSim. These include *system configuration requirements* and *user requirements*.

Configuration Requirements

NightSim requires the Frequency Based Scheduler to be installed on the target system. The FBS is an optional portion of RedHawk Linux.

Typically, the default installation of the FBS is sufficient for NightSim activities. Refer to the *RedHawk Frequency-Based Scheduler (FBS) User's Guide (0898005)* for more information on FBS configuration.

NightSim schedulers that are driven by an interrupt source require an RCIM device; a real-time clock, an edge-triggered interrupt, or a distributed interrupt source. If you do not have an RCIM device, you can still use NightSim's end-of-process scheduling capability to drive your simulation. See *End of Cycle* scheduling for more information.

User Requirements

To use NightSim, you must also ensure that certain user requirements related to remote shell access, environment variables, and capabilities are met. These requirements are described in the sections that follow.

Remote System Access

The secure shell, **ssh(1)**, is only required if you use NightSim to control a scheduler on a remote target. It is used to authenticate your user ID on the target system and to initiate execution of the NightSim server process.

When an **xterm** option is specified for a process on a remote target in the scheduler definition, NightSim will run the **xterm** command on the NightSim target and rely on X11-forwarding through the established ssh connection to show the xterm on the host system.

Capabilities

Some of the operations associated with the *frequency-based scheduler* and the *performance monitor* require certain capabilities such as `CAP_SYS_NICE`.

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 `fbscheduser` role which provides the `CAP_SYS_NICE` capability.

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

```
role fbscheduser CAP_SYS_NICE
```

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

```
user username fbscheduser
```

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

If the user requires capabilities not defined in the `fbscheduser` role, add a new role which contains `fbscheduser` and the additional capabilities needed, and substitute the new role name for `fbscheduser` 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. Table 2-1 presents a recommended configuration that will grant capabilities to users of the services most commonly employed in accessing a system.

Table 2-1. Recommended /etc/pam.d Configuration

/etc/pam.d File	Affected Services	Comment
remote	telnet rlogin rsh (when used <u>w/o</u> 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 <u>w/o</u> 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 <u>with</u> 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 restart 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.

This chapter explains the procedures for beginning and ending a NightSim session. It also explains how to get help and how to use accelerators.

Invoking NightSim

The NightSim tool is available on your system as `/usr/bin/nsim`.

The syntax for executing `nsim` is as follows:

```
nsim [-h|--help] [-v|--version]
      --monitor
      --scheduler=[targetname,]key]
      --target=targetname]
      [-Xoption ...] [--config=file]
```

Options are described as follows:

--help

Allows you to display the usage information for NightSim and then exit.

--version

Allows you to display the version and copyright information for NightSim and then exit.

--monitor

Starts NightSim with the Monitor page raised. This option can be abbreviated as **-m**.

--scheduler=[targetname,]key

Configures a scheduler associated with an FBS with the key *key* on the system *targetname*. If *targetname* is not specified, the window will be associated with an FBS on the local system where `nsim` is invoked.

key is an integer value that identifies the scheduler.

targetname can be of the form *user@target_system*, which allows you to specify the remote system user name.

This option can be abbreviated as **-s** *[targetname,]key*.

--target=*targetname*

Configures a new scheduler on system *targetname*.

targetname can be of the form *user@target_system*, which allows you to specify the remote system user name.

This option can be abbreviated as **-t** *targetname*.

-Xoption

Specifies any standard X Toolkit command-line option.

[--config=]*file*

Allows you to specify the name of a file that contains configuration and scheduling data for a selected *frequency-based scheduler*.

To save the configuration and scheduling data entered in a particular Scheduler page to a file, select the Scheduler->Save Config As menu item. See Save Config As on page 3-4 for details.

This option can be abbreviated as **-c** *file*, or simply *file* (if it is the last argument).

Options can be specified in any order.

If both a scheduler key and a configuration file are supplied, the target system, user, scheduler, and process information from the configuration file are ignored in favor of the specified active scheduler. Other information from the configuration file (such as window and panel geometry, and table column settings and sort order) are utilized.

You may also invoke NightSim without specifying any options. Doing so allows you to display an inactive window. An inactive window is one that is not associated with a scheduler that exists on the system.

The steps for invoking NightSim are as follows:

1. Log in to your system.
2. Ensure that the value of your DISPLAY variable is set to the name of your X server (or use the **-display** command-line option to the NightSim command).
3. Type **nsim** and any desired options after the system command prompt, and press the Enter key.

The Menu Bar

The menu bar provides access to configuration settings, tools, scheduler operations, obtaining help, and controlling which areas are displayed in the window.

The menu bar provides the following menus:

- File
- Target
- View
- Scheduler
- Monitor
- Tools
- Help

Each menu is described in the sections that follow.

File Menu

Mnemonic: Alt+F

The File menu allows you to load or save configuration data and contains the means to exit NightSim.

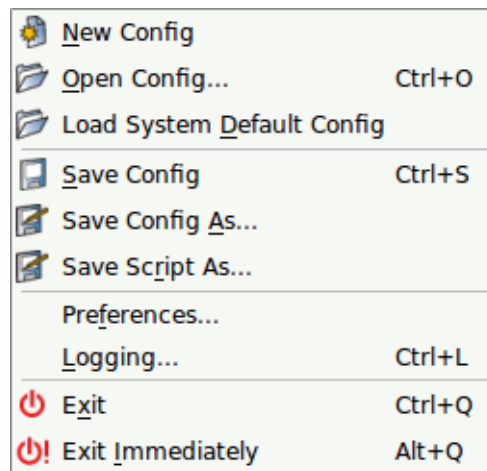


Figure 3-1. File Menu

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

New Config

Mnemonic: N

This option clears all information from the current window, resets the various areas to blank or default values and places the window in the inactive state.

Open Config...

Mnemonic: O

Accelerator: Ctrl+O

This option allows you to open a scheduler configuration file that you have previously saved.

To assist you in specifying a configuration file to be opened, NightSim displays a file selection dialog. After making a selection, you may open the selected file, search for another file or cancel the operation. When you open the selected file, the Scheduler window is associated with that file and displays the configuration and scheduling data contained in the file.

Load System Default Config

Mnemonic: D

This option clears all information from the current window, resets the various areas to blank or default values and loads the default system configuration file. All window geometries and defaults are reset as per the configuration file.

The default configuration file is also consulted for all invocations of NightSim on the system that do not specify a specific configuration file.

If you wish to make adjustments to the default configuration file, load it, make your adjustments, and then save it to a local directory. Then copy the local file to `/usr/lib/NightSim-version/default_config`. The value of `version` changes from release to release. Updating the file in the path mentioned above requires root access.

Save Config

Mnemonic: S

Accelerator: Ctrl+S

This option allows you to save the configuration and scheduling data entered in the current window in the file that is associated with the window. If the configuration file name has not yet been specified, this option is disabled; you should use the Save Config As... option.

Save Config As...

Mnemonic: A

This option allows you to specify the name of the file in which you wish the configuration and scheduling data entered in the current window to be saved.

When you select this option, NightSim displays a file selection dialog. You may save the current configuration and scheduling data in a selected file, name a new config file, search for another file or cancel the operation.

Save Script As...

Mnemonic: R

This option allows you to save the current configuration as a shell script containing **rtcp** commands. The script may be run from any host in your network as long as it has **ssh(1)** access to the target system (see “Remote System Access” on page 2-1). This script can then be used to recreate your scheduler configuration without the need of an X11 display.

The main disadvantage of using this script instead of using NightSim is that you do not get any monitoring capabilities. This may be less important for systems that are typically used in production environments but can be critical to those systems used in a testing/development environment where the monitoring feedback that NightSim provides can be invaluable.

Preferences...

Mnemonic: F

This option launches the **Preferences Dialog** which allows you to specify preferences for NightSim, including font selection.

Saved user preferences are applied to all NightSim invocations for the user. Preferences are saved in the user’s home directory and have a broader application than session configuration files.

See “Preferences Dialog” on page 3-14 for more information.

Logging...

Mnemonic: L

Accelerator: Ctrl+L

This option launches the **Logging Dialog** which allows you to enable and disable performance monitor value logging and to select the values to be logged.

See “Logging” on page 5-21 for more information.

Exit

Mnemonic: X

Accelerator: Ctrl+Q

This option allows you to exit NightSim. If you have an active scheduler, NightSim displays a dialog verifying that you want to exit, warning that the schedulers will remain active if you exit.

If you confirm that you want to exit, NightSim then checks whether there are changes to the current configuration which have not yet been saved. If so, you will

be asked if you wish to save the changes. If you click on the **Yes** button to save the changes, NightSim saves the current configuration and then exits. If the current configuration is unnamed, a file selection dialog will be displayed. If you click on **No**, NightSim exits without saving the current configuration. Clicking on **Cancel** cancels the exit.

If there are no active schedulers set up and/or running and no outstanding changes to be written to the config file, NightSim exits immediately.

Exit Immediately

Mnemonic: I
Accelerator: Alt+Q

This option exits NightSim immediately without the verification and checks performed by the Exit option.

Target Menu

Mnemonic: Alt+T

The **Target** menu allows you to select a new target system.

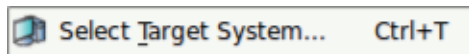


Figure 3-2. Target Menu

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

Select Target System...

Mnemonic: T
Accelerator: Ctrl+T

This menu option presents a target system selection dialog which allows you to specify the target system, the remote user ID, and any runtime scheduling attributes desired for the NightSim server process.

The runtime scheduling attributes you set in that dialog only apply to the NightSim server process, not to processes you schedule on an FBS via NightSim.

However, you should use real-time priorities and scheduling classes for the NightSim server so that it can make adjustments to and launch additional processes on heavily loaded systems. Alternatively, you can ensure that the server runs on a CPU which is not part of your real-time application suite.

View Menu

Mnemonic: Alt+V

The View menu allows you to control the tools and areas displayed in the current Night-Sim session.

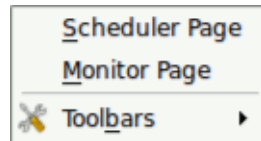


Figure 3-3. View Menu

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

Scheduler Page

Mnemonic: S

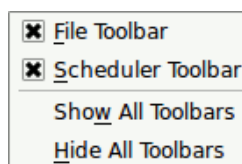
This menu item raises the Scheduler Page. The Scheduler page is described in detail in “Using the Scheduler Area” on page 4-1.

Monitor Page

Mnemonic: M

This menu item raises the Monitor Page. The Monitor page is described in detail in “Using the Monitor Area” on page 5-1.

Toolbars



Mnemonic: B

This menu item allows control of which tools are displayed in the toolbar. The toolbar is discussed in more detail in the section “The Toolbar” on page 3-12.

Scheduler Menu

Mnemonic: Alt+S

The Scheduler menu allows you to control the scheduler.



Figure 3-4. Scheduler Menu

This menu is discussed in the section “The Scheduler Menu” on page 4-5.

Monitor Menu

Mnemonic: Alt+M

The Monitor menu contains general commands related to performance monitoring.

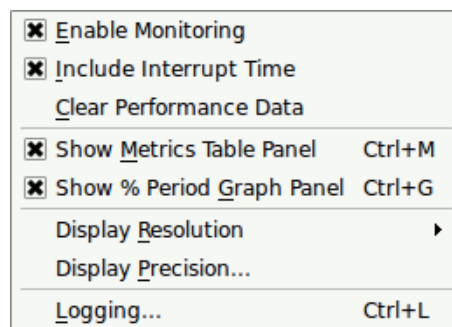


Figure 3-5. Monitor Menu

This menu is discussed in the section “The Monitor Menu” on page 5-9.

Tools Menu

Mnemonic: Alt+T

The **Tools** menu contains a list of other NightStar tools that can be launched directly from NightSim.

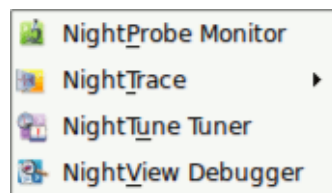


Figure 3-6. Tools Menu

Descriptions of the options on the **TOOLS** menu follow.

NightProbe Monitor

Mnemonic: P

Opens the NightProbe Data Monitoring application.

NightProbe is a real-time graphical tool for monitoring, recording, and altering program data within one or more executing programs without significant intrusion. NightProbe can be used in a development environment as a tool for debugging, or in a production environment to create a “control panel” for program input and output.

See the *NightProbe RT User’s Guide* for more information.

NightTrace

Mnemonic: T

Displays a sub-menu which allows you to launch the NightTrace Analyzer or the NightTrace Illuminator.

The NightTrace Analyzer is a graphical tool for analyzing the dynamic behavior of multiprocess and/or multiprocessor 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.

The NightTrace Illuminator is a graphical tool which automatically instruments programs, libraries, or modules by simply reading the linked or compiled form. It provides trace events on the entry and exit of all functions for subsequent analysis with the NightTrace Analyzer. No source changes are required and the Illuminator allows you to control which functions are to be instrumented.

See the *NightTrace RT User’s Guide* for more information.

NightTune Tuner

Mnemonic: U

Opens the NightTune Performance Tuner.

NightTune is a graphical tool for analyzing the status of the system in terms of processes, interrupts, context switches, interrupt CPU affinity, processor shielding and hyperthreading control as well as network and disk activity. NightTune can adjust the scheduling attributes of individual or groups of processes, including priority, policy, and CPU affinity.

See the *NightTune RT User's Guide* for more information.

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 the *NightView RT User's Guide* for more information.

Help Menu

Mnemonic: Alt+H

NightSim provides on-line help information through the Help menu. Figure 3-1 presents this menu:

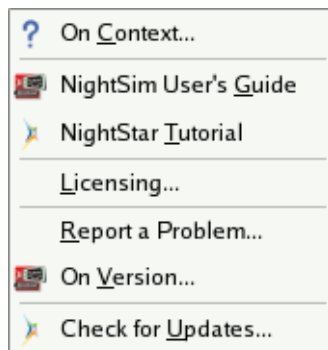


Figure 3-1. Help Menu

The following describe the options on the Help menu:

On Context...

Mnemonic: C

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

Help for a particular item is obtained by first choosing this menu option, then clicking the mouse pointer on the object for which help is desired (the mouse pointer will become a floating question mark when the **On Context** menu item is selected).

In addition, context-sensitive help may be obtained for the currently highlighted widget by pressing the F1 key. QT-Assistant, NightSim's online help system, will open with the appropriate topic displayed.

NightSim User's Guide

Mnemonic: G

Opens the online version of the *NightSim RT User's Guide* in the online help viewer.

NightStar Tutorial

Mnemonic: T

Opens an online help window containing a tutorial which demonstrates the features of NightSim™, NightProbe™, NightView™, NightTrace™ and NightTune™ in one cohesive example.

License Report...

Mnemonic: L

Provides licensing information.

Report a Problem...

Mnemonic: R

Launches a dialog with a hyperlink to Concurrent's web site where you can enter a problem report. The dialog also contains other contact information.

On Version

Mnemonic: V

Displays a short description of the current version of NightSim.

Check for Updates...

Mnemonic: U

Launches NUU (Network Update Utility) enabling you to update your system. This requires network access to Concurrent's Updates web site. Updates require a login and user ID issued by Concurrent. Refer to <http://redhawk.ccur.com/updates> for complete information.

This menu item is not shown on Debian-based systems. Use the standard Debian update tools to obtain NightSim updates. See the URL shown in the paragraph above for more information.

NightSim also provides help information through the Help buttons that appear in dialog windows. When you click on the Help button in one of these windows, NightSim displays the help information for that window.

The Toolbar

The NightSim toolbar provides icons for commonly used actions.



Figure 3-2. NightSim Toolbar

File Toolbar:



Configures a new NightSim scheduler



Opens an existing NightSim configuration. See **Open Config** on page 3-4 for details.



Saves the current NightSim scheduler configuration. See **Save Config** on page 3-4 for details.

Scheduler Toolbar:



Selects a scheduler that is active on the target system. See **Select Active Scheduler** for details.



Sets up the configured scheduler. See **Setup Scheduler** on page 4-7 for details.



Toggles between start and restart a scheduler, resetting the frame and cycle counters to zero. See **Start/Restart Scheduler** on page 4-7 for details.



Toggles between pause and resume the scheduler. See **Pause Scheduler** on page 4-8 and **Resume Scheduler** for details.



Removes a configured scheduler. See **Remove Scheduler** on page 4-8 for details

Using Accelerators

A set of keyboard accelerators is defined for use within NightSim to provide quick access to certain functions and menu items. Accelerators associated with menu items allow you to access those items whether or not the menu is posted. All menus can be activated by using the **Alt** key in addition to the first letter of the menu. The following table contains a list of additional accelerators and the resulting actions.

Table 3-1. Accelerator Table

Key Sequence	Action
Alt+A, Ctrl+N	Add a new process to the process list. Launches the Add Process dialog.
Alt+C, Ctrl+C	Copies the selected process in the process list and inserts an identical process into the list.
Alt+D	Raises the Scheduler page if it is not already raised.
Alt+E, Ctrl+E	Launches the Edit Process dialog for the process selected in the process list.
Alt+O	Raises the Monitor page if it is not already raised.
Alt+Q	Exits NightSim immediately. If a scheduler is active, it is left in the active state and NightSim exists without warning the user.
Alt+R	Refreshes the process list by querying the actual scheduler on the target system. NightSim automatically does a refresh for you every 10 seconds while the scheduler is active.
Ctrl+L	Launches the Logging dialog.
Ctrl+M	Toggles the visibility of the Metrics Table panel on the Monitor page. If the panel becomes visible, the Monitor page is raised as well.
Ctrl+O	Opens a file selection dialog for you to specify a configuration file to load.

Table 3-1. Accelerator Table

Key Sequence	Action
Ctrl+P	Toggles the visibility of the Percent Period Graph panel on the Monitor page. If the panel becomes visible, the Monitor page is raised as well.
Ctrl+Q	Initiates exit processing for NightSim. If a scheduler is active, NightSim first asks you whether it should remain active before exiting. You can use Alt+Q to cause NightSim to exit immediately, regardless of whether an active scheduler exists or not.
Ctrl+R	When the Monitor page is raised, this accelerator rotates the Metrics Table . The table operates in two modes. By default, processes are shown in rows with the columns containing process attributes. When rotated, the columns contain the processes and the rows contain the process attributes.
Ctrl+S	Saves the current configuration file. If no configuration file has yet been specified, NightSim automatically selects a new file name in the current directory and saves the file -- such files have a .nsc suffix. The name of the configuration file is shown in the status bar at the bottom left of the main window.
Ctrl+T	Launches the target system selection dialog.
Ctrl+Shift+P	Pauses the currently active scheduler.
Ctrl+Shift+P	Pauses the current scheduler.
Ctrl+Shift+R	Resumes the current scheduler -- frame and cycle values continue from their previous values (when the scheduler was last paused).
Ctrl+Shift+S	Starts the active scheduler (or restarts it from a paused state). Frame and cycle values will reset to zero.
Ctrl+Shift+U	Sets up the scheduler and causes it to become active (programs are launched and the interrupt device, if any, is programmed and attached to the scheduler).
F1	Invokes the integrated help system and displays the section of the Night-Sim User's Guide specific to the widget or page that has focus.

Preferences Dialog

The Preferences Dialog is launched via the **Preferences...** option of the **File** menu.

The dialog has two tabbed pages; **General** and **Fonts**, as described in the sections that follow.

The buttons at the bottom of the page control the application of your changes.

OK

Applies any changes made in the dialog to the current invocation of NightSim and closes the dialog. The changes will not be saved to disk or affect subsequent NightSim invocations unless you return to the **Preferences...** dialog and press **Save**.

Reset

Discards any changes you have made directly to the dialog since the dialog was launched and resets the it accordingly.

Note:

Changes made in the **NightTrace Global Fonts** dialog cannot be discarded via the **Reset** button.

Save

Applies the preferences from the dialog to the current invocation of NightSim, saves the preferences to disk thereby affecting subsequent NightSim invocations, and closes the dialog.

Cancel

Cancels any pending changes and closes the dialog.

Help

Opens the help system to display this section.

Separation of apply (**OK**) and **Save** operations make it easy to experiment with preference changes in the current invocation without affecting long-term usage.

General Preference Page

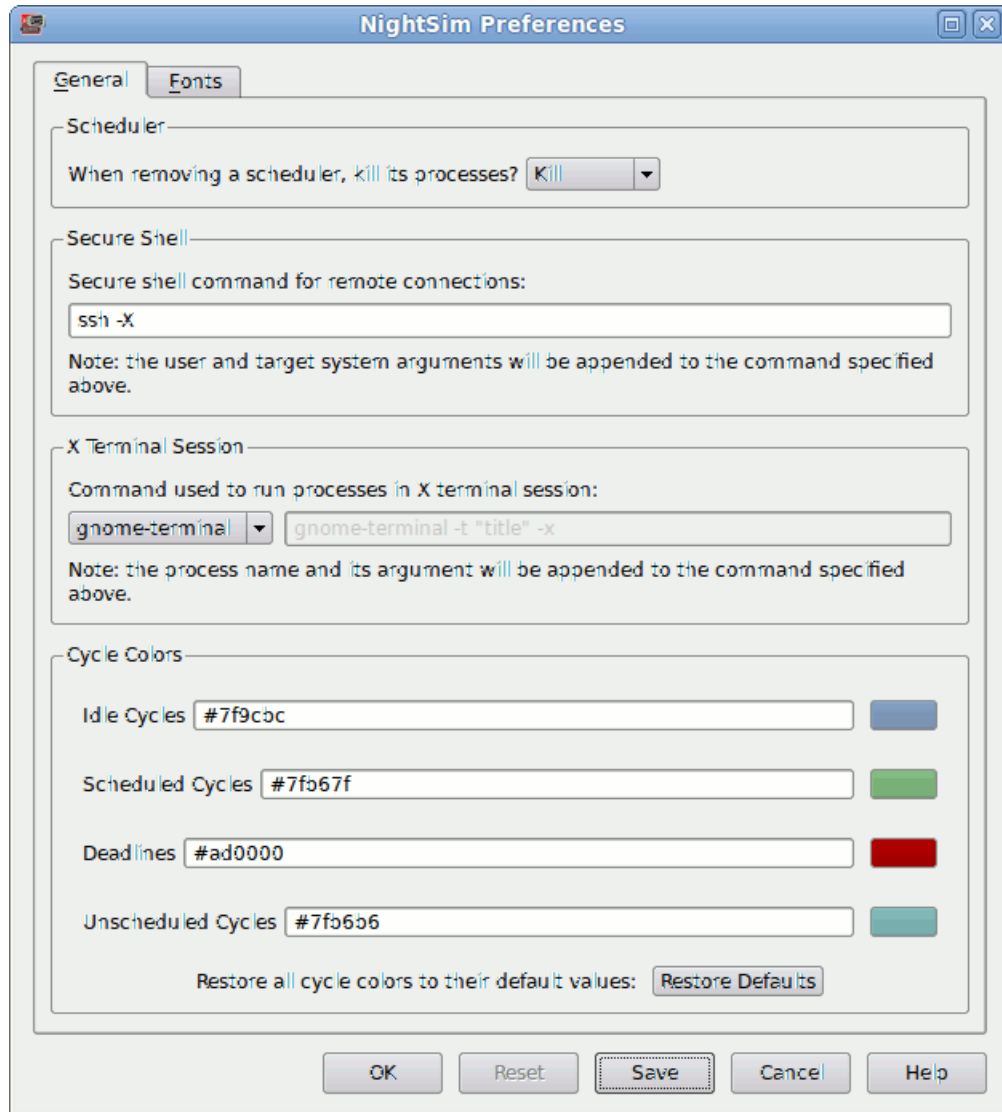


Figure 3-3. General Preference Page

The page is divided into four sections.

Scheduler

The area allows you to specify NightSim's actions when removing a scheduler. The option list provides three choices:

- Kill -- NightSim will automatically kill all processes on the scheduler without asking.

- **Don't Kill** -- NightSim will remove the scheduler, but its processes will continue to execute. Note however that a typical coding style for applications using the FBS is to exit after `fbswait(3)` returns a -1 value. `fbswait(3)` will return with -1 (and `errno` set to `ENOENT`) after the scheduler has been removed.
- **Ask** -- NightSim will first ask you if you want to kill the processes or leave them running.

Secure Shell

A secure shell is used to make connection to remove targets (and to launch the NightSim server process). You can specify the path to the `ssh` command and supply options. If you specify a program other than `ssh`, NightSim may not function.

Do not specify options or arguments for the remote user or remote system name; these will be appended by NightSim to the command as specified in this area.

It is important to ensure that X11 forwarding is allowed, as that is required for running an FBS process inside an X terminal session. By default, this is specified (the `-X` option).

X Terminal Session

When selecting to run an FBS process inside an X terminal session (so that you can interact with the process or perhaps because the process has specialized I/O requirements), the command as specified in this section invoked from the target system.

By default, the command will use the display setting of the existing `ssh` connection, however you can override that value in the **Add Process** dialog. Do not set the display value in the **X Terminal Session** command.

When specifying a custom command, ensure that any options that need to precede process commands and arguments are specified last. NightSim will append to the specified command the name of the process to invoke and any options it requires.

Cycle Colors

This area allows you to customize the colors used to display the scheduling cycles for a process. It affects the **Schedule** column of the **Process Table** as well as the **Key** (legend) icons at the bottom of that table. See "Using the Processes Area" on page 4-15 for more information.

Fonts Preference Page

NightSim uses multiple fonts to present text in the most effective manner throughout the various display areas of the tool.

Variable-width fonts are most commonly used; these fonts most closely resemble how people write or print words.

Fixed-width fonts require that all characters and numbers have the same width (visual footprint). Fixed-width fonts are of benefit when source code is being displayed or manipulated or when columns of numbers are viewed.

NightSim further divides the use of fonts into the following categories; default and panel.

Default fonts are used for text associated with operational description and control, including menus, buttons, selection devices, labels, tool tips, status bar messages, and generally descriptive verbiage.

Panel fonts are used in NightSim panels, which display the data of highest importance to you.

Fonts are selected by querying font preferences from the following sources until a preference is found:

- Your NightSim preference
- Your NightStar-wide preference
- The system's NightSim preference
- The system's NightStar-wide preference
- NightSim's ultimate default

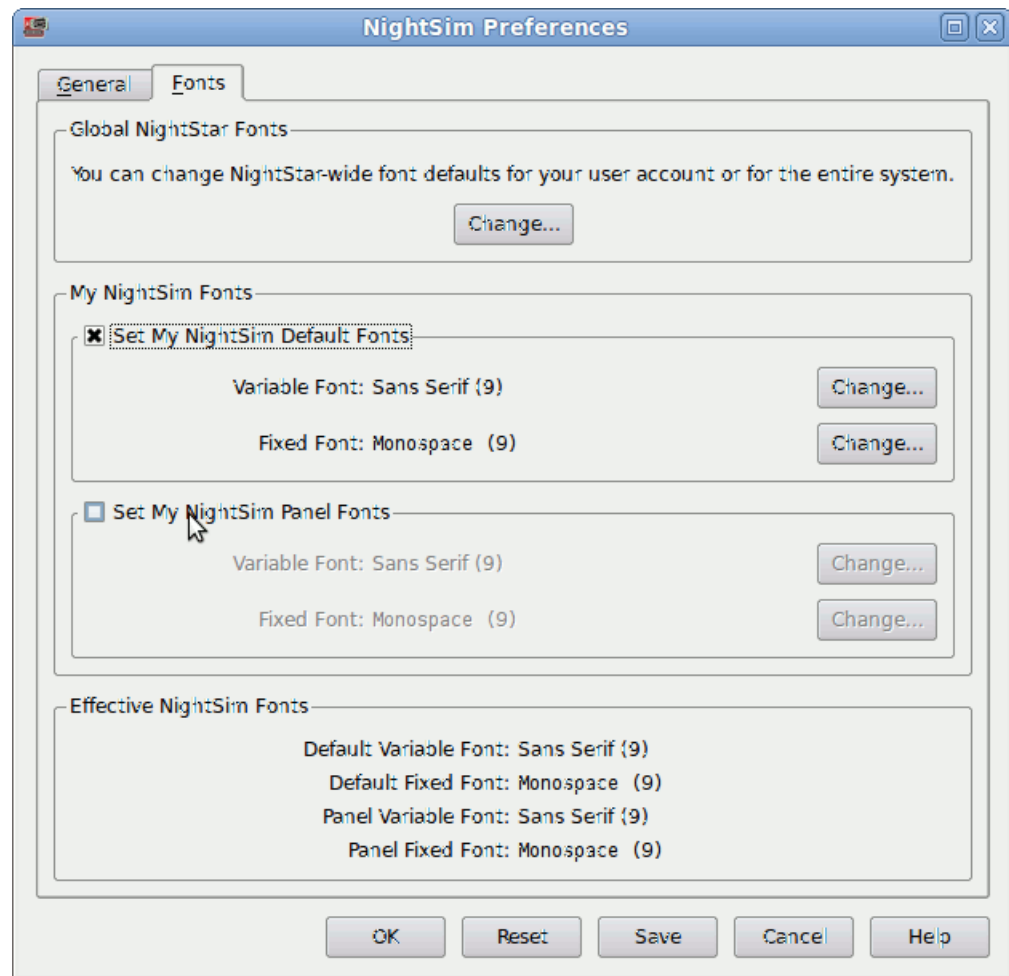


Figure 3-4. Fonts Preferences Page

This page is divided into three sections.

Global NightStar Fonts

The **Change...** button in this area launches the **NightStar Global Fonts** dialog which allows you to set your Nightstar-wide preferences, your preferences for another specific NightStar tool, or the system's tool or NightStar-wide preferences.

Note:

Setting a NightStar preference for the system typically requires root access.

Changes saved in the **NightStar Global Fonts** dialog are always saved to disk and apply to the current and subsequent NightSim invocations.

My NightSim Fonts

This area allows you to set or clear your user's preferences for NightSim.

Selection of the checkboxes for the individual font categories control whether or not your preferences are to be consulted. Clearing a checkbox effectively removes your user preference for that category. Setting a checkbox allows you to select specific fonts within the category.

Changes to any of the settings in this area, including individual fonts or category checkboxes, are immediately reflected in the **Effective NightSim Fonts** area at the bottom of the page so you can see the ultimate effect a change will have.

To change a specific font, ensure that the corresponding category's checkbox is checked and then press the **Change...** button. This will launch a standard font selection dialog. When you select a font from the dialog and press **OK**, the name of the font family is displayed to the left of the **Change...** button and is displayed in the selected font as well.

Effective NightSim Fonts

This area shows you the effective fonts that will be used, based on your user settings and consultation of global settings which aren't shown in the page.

The values in this area immediately change to reflect the effective font whenever any change is made within the page.

As with other NightSim preferences, your changes in the **My NightSim Fonts** area are applied to the current invocation of NightSim when you press the **OK** or **Apply** buttons.

However, your changes are not saved to disk and will not affect subsequent invocations of NightSim unless you press the **Save** button.

Note:

Changes to font preferences in the **NightStar Global Fonts** dialog are always saved to disk and apply to the current and subsequent NightSim invocations; i.e. there is no way to experiment with a global font preference without affecting subsequent NightSim invocations.

NightStar Global Fonts Dialog

The NightStar Global Fonts dialog allows you to set your Nightstar-wide preferences, your preferences for another specific NightStar tool, or the system's tool or NightStar-wide preferences.

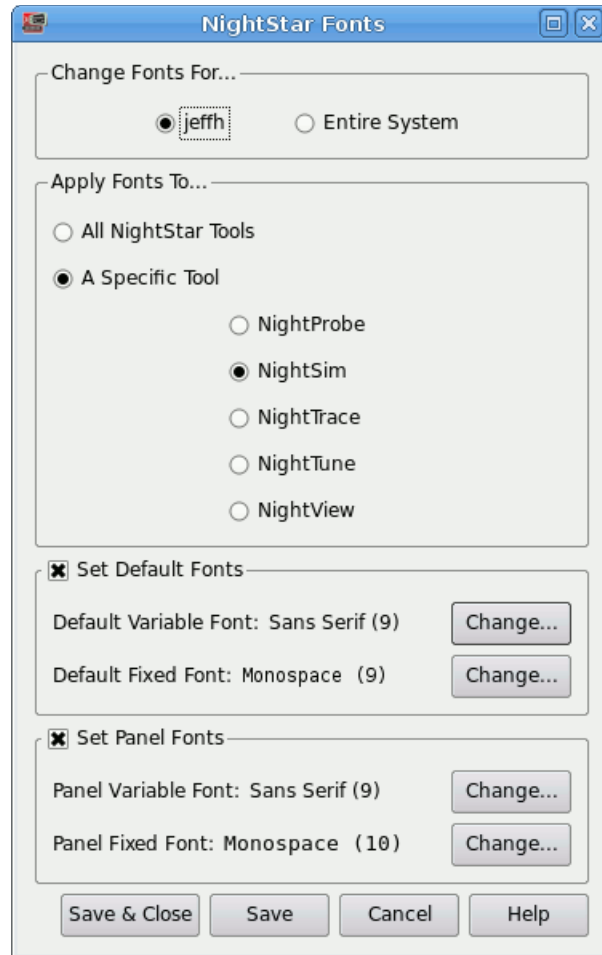


Figure 3-5. NightStar Global Fonts Dialog

Keep in mind that fonts are selected by querying font preferences from the following sources until a preference is found:

- Your NightSim preference
- Your NightStar-wide preference
- The system's NightSim preference
- The system's NightStar-wide preference
- NightSim's ultimate default

This dialog has two control areas which define the scope of font preference application.

Changes Fonts For...

By default, the dialog is set up to apply font preferences to your user account. Select the **Entire System** button if you wish to set the system's preferences.

Note:

Changing font preference for the system typically requires `root` access.

Apply Fonts To...

This area additionally controls the scope of font preference application. You can change a preference for a specific NightStar tool or change the NightStar-wide preference.

If you wish to change the font for more than one tool from this dialog, but not change the NightStar-wide preference, select the first tool of interest, make your preference change in the areas below, and then press the **Save** button. Then select the second tool of interest and repeat.

Set Default Fonts Set Panel Fonts

These areas contain the variable and fixed-width font preferences for each of the font categories, identified by the label next to each checkbox.

To remove the preferences in a category, clear its checkbox.

To change a specific font, ensure that the category's checkbox is checked and then press the **Change...** button. This will launch a standard font selection dialog. When you select a font from the dialog and press **OK**, the name of the font family is displayed to the left of the **Change...** button and is displayed in the selected font as well.

The buttons at the bottom of the page control the application of your changes.

Save & Close

Saves any changes made in this dialog to disk, thus affecting subsequent tool invocations, and closes the dialog.

These changes may affect the effective font preferences for the current invocation of NightSim. When the dialog is closed, the fonts shown in the **Effective NightSim Fonts** section of the **Preferences** dialog are updated. If you apply the changes in that dialog, they will take effect in the current invocation of NightSim.

Save

Applies the preferences from the dialog to the current invocation of NightSim, saves the preferences to disk thereby affecting subsequent NightSim invocations.

These changes may affect the effective font preferences for the current invocation of NightSim. When this dialog is subsequently closed, the fonts shown in the **Effective NightSim Fonts** section of the **Preferences** dialog are updated. If you apply the changes in that dialog, they will take effect in the current invocation of NightSim.

Cancel

Cancels any unsaved changes and closes the dialog.

Help

Opens the help system to display this section.

Exiting NightSim

You may exit NightSim by using one of the following techniques:

- Select the **File->Exit** menu item or press **Ctrl+Q**. If schedulers are active or the current configuration is not yet saved, you will be asked to confirm that you wish to exit.
- Select the **File->Exit Immediately** menu item or press **Alt+Q**. NightSim will exit without checking for active schedulers or unsaved configurations and does not ask for verification.

Using the Scheduler Area

This chapter describes the *frequency-based scheduler* and explains how scheduler frequency is defined and how processes are scheduled. It introduces the components of the Scheduler area of the NightSim window and describes the use of each component.

Understanding the Frequency-Based Scheduler

The *frequency-based scheduler* is a task synchronization mechanism that allows you to run processes at frequencies that you specify. Frequencies can be based on high-resolution clocks, an external interrupt source, or completion of a cycle. The frequency-based scheduler provides a mechanism for initiating processes at the specified frequency, and optionally with a specified deadline.

The frequency-based scheduler provides you with the ability to:

- Define *FBS* frequency in terms of the duration of a *minor cycle* and the number of minor cycles per *major frame*
- Specify the scheduling parameters with which processes are scheduled
- Detect *deadline violations* for FBS-scheduled processes having assigned deadlines
- Detect *frame overruns* for all FBS-scheduled processes
- View the status of FBS-scheduled processes
- Remove one or all FBS-scheduled processes from a scheduler
- Reschedule an FBS-scheduled process
- Start, stop, and resume scheduling on a frequency-based scheduler
- Connect an interrupt source to and disconnect it from a frequency-based scheduler
- Control use of the real-time clock device as the interrupt source for a frequency-based scheduler
- Configure up to 100 frequency-based schedulers system-wide in a single processor or multiprocessor environment
- Use both frequency-based scheduling and static priority scheduling simultaneously

In addition, NightSim provides the following additional capabilities for your convenience:

- Specify input and output files for scheduled processes

- Specify a working directory for scheduled processes
- Schedule processes under **xterm(1)** terminal windows, providing the option of interactive behavior
- Schedule processes under the NightView debugger
- Specify deadlines in terms of cycle counts when using a Clock Interrupt Source

NOTE

Some of these additional features are not available for schedulers started via a saved script.

How Is Scheduler Frequency Defined?

You configure a *frequency-based scheduler*, in part, by defining the number of *minor cycles* that compose a *major frame*. Minor cycles and major frames have associated with them a duration of time that you can define by using an interrupt source for the scheduler. The interrupt source can be the end of a minor cycle, a real-time clock, an edge-triggered interrupt, or a user-supplied device.

If you use *end-of-cycle scheduling*, scheduling is triggered when the last process that is scheduled during the current minor cycle of the current major frame completes its processing.

If you use a real-time clock as the interrupt source, you can specify the period in various units of time or the frequency in hertz.

How Are Processes Scheduled?

You schedule processes to run at a certain frequency by specifying the first *minor cycle* in which the process is to be wakened in each *major frame* (called the *starting base cycle*) and the frequency with which it is to be wakened (called the *period*).

If, for example, you schedule “Process 1” with a starting base cycle of zero and a *period* of two, the process will be wakened once every two minor cycles, starting with the first minor cycle in the frame.

If you schedule “Process 2” with a starting base cycle of one and a period of four, that process will be wakened once every four minor cycles, starting with the second minor cycle in the frame.

If you then schedule “Process 3” with a starting base cycle of two and a period of two, that process will be wakened once every two minor cycles, starting with the third minor cycle in the frame.

On a *frequency-based scheduler* configured with 100 minor cycles per major frame, these processes will be wakened as illustrated in Table 4-1.

Table 4-1. Process Scheduling

Minor Cycle	Processes Wakened
0	Process 1
1	Process 2
2	Process 1, Process 3
3	
4	Process 1, Process 3
5	Process 2
...	
97	Process 2
98	Process 1, Process 3
99	

The maximum frequency with which you can schedule a process is once per minor cycle (a period of one); the minimum frequency is once per major frame (in the case of the example, a period of 100).

A process runs until it calls an *FBS* library routine that causes it to sleep until the frequency-based scheduler wakes it again. The frequency-based scheduler wakes those sleeping processes that are scheduled to be wakened in the current minor cycle of the current major frame and repeats the process for each minor cycle in the current frame. It continues to repeat the entire process on every major frame until the scheduler is disabled. A scheduler configured with 100 minor cycles per major frame, a minor cycle duration of 10,000 microseconds (0.01 second), and a major frame duration of one second wakes processes as illustrated in Table 4-2.

Table 4-2. Scheduler Operation

Major Frame	Time (sec.)	Minor Cycle	Processes Wakened
0	0	0	Process 1
	0.01	1	Process 2
	0.02	2	Process 1, Process 3
	...		
	0.97	97	Process 2
	0.98	98	Process 1, Process 3
	0.99	99	

Table 4-2. Scheduler Operation (Cont.)

Major Frame	Time (sec.)	Minor Cycle	Processes Wakened
1	1.00	0	Process 1
	1.01	1	Process 2
	1.02	2	Process 1, Process 3
	...		
	1.97	97	Process 2
	1.98	98	Process 1, Process 3
	1.99	99	
	...		
n	n.00	0	Process 1
	n.01	1	Process 2
	n.02	2	Process 1, Process 3
	..		
	n.97	97	Process 2
	n.98	98	Process 1, Process 3
	n.99	99	

As illustrated in Table 4-2:

- When the current major frame is zero and the current minor cycle is zero, the scheduler wakes “Process 1.”
- After 0.01 second, it wakes “Process 2”; after 0.02 second, it wakes “Process 1” and “Process 3”; and so on.
- At one second, when the current major frame becomes one, the current minor cycle becomes zero again, and the scheduler wakes “Process 1.”
- After 0.01 second, it wakes “Process 2.”
- After 0.02 second, it wakes “Process 1” and “Process 3”; and so on.
- The scheduler continues repeating this process for as long as it is enabled.

It is important to note that a process may not always run at the frequency that you have specified. A *frame overrun* occurs when a scheduled process does not finish its processing before it is scheduled to run again. Overrun counts are maintained for each process and for each scheduler. You can obtain information on overruns by querying a process in the Monitor area (see “Using the Monitor Area” on page 5-1). When you schedule a process on a selected scheduler, you have the option of ensuring that the scheduler is stopped when that

process causes a frame overrun. The “Halt on Overrun” feature may be used to ensure that the scheduler is stopped when a frame overrun occurs.

If a deadline has been specified for a scheduled process and your process does not finish its processing before the deadline expires, a *deadline violation* occurs. NightSim’s **Halt on Deadline Violation** option allows you to ensure that the scheduler is stopped when a deadline violation is detected.

For additional information on the *frequency-based scheduler*, refer to the *RedHawk Frequency-Based Scheduler (FBS) User’s Guide (0898005)*.

Programming Requirements

Programs scheduled to run on a frequency-based scheduler have certain requirements:

- The program must call **fbwait(3)**.
- The program should complete its work unit and return to **fbwait** before its next scheduled cycle or deadline occurs; otherwise an overrun or deadline violation may occur (whose handling is configurable).

Refer to the *RedHawk Linux Frequency-Based Scheduler User’s Guide*, publication number 0898005, for more information about FBS programming.

The Scheduler Menu

Scheduler control can be performed using selections from the Scheduler Menu of the NightSim window menu bar (see “The Menu Bar” on page 3-3). It can also be performed using the Scheduler Control area (see “Using the Scheduler Control Area” on page 4-9) and tools on the NightSim toolbar (see “The Toolbar” on page 3-12).

The following figure illustrates NightSim’s Scheduler menu.

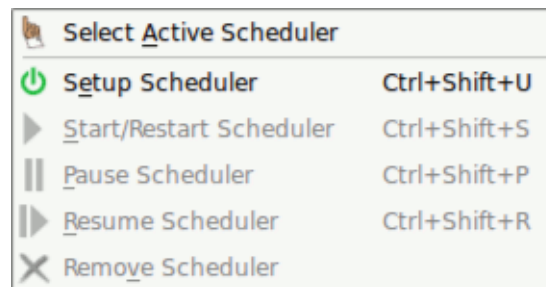


Figure 4-1. Scheduler Menu

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

Select Active Scheduler

Mnemonic: A

This option allows you to associate the current NightSim window with a *frequency-based scheduler* active on a given *target system*. When you select this option, NightSim displays Figure 4-2.

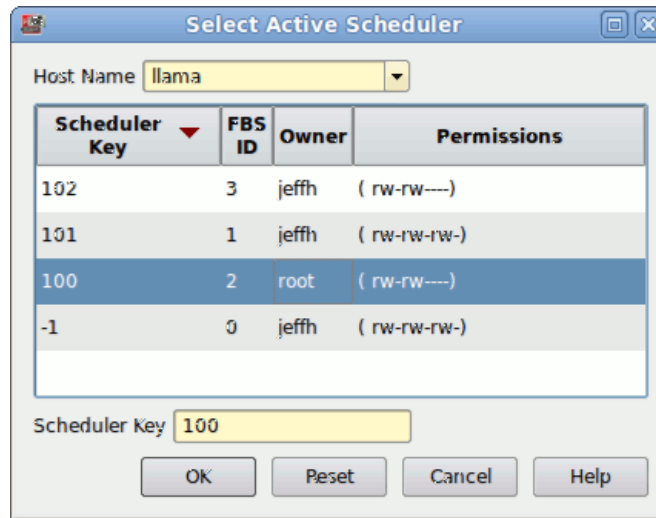


Figure 4-2. Select Active Scheduler Dialog


This dialog provides a list of the *active schedulers* for the system designated by the **Host Name** field. The **Host Name** defaults to your local host and the list contains the active schedulers running on that particular system. For instance, Figure 4-2 lists the three schedulers currently running on the host llama.

You may change the **Host Name** either by using the drop-down list or by typing the name of the *target system* into the text field provided. Entering an item in this field will populate the list with the *active schedulers* on the specified target.

You may select a scheduler from the list that the dialog displays, or you may use the **Scheduler Key** text field to enter the numeric *key* that is associated with the desired scheduler. The key can be any positive integer value. Whether you select from the list or enter a key, the selected scheduler must previously have been configured, and you must have at least read permission for it; otherwise, NightSim reports an error.

After making your selections, click **OK** to apply the selections and close the dialog, **Reset** to reset original values, **Cancel** to discard selections and close the dialog, or **Help** to display help.

If you successfully select an existing scheduler, the NightSim window becomes an active window associated with that scheduler. Information related to the selected scheduler replaces information that the window contained prior to the operation.

The **Select Active Scheduler** option is equivalent to using the  icon on the scheduler toolbar (see “The Toolbar” on page 3-12).

Setup Scheduler

Mnemonic: E


This menu item is used to set up the configured scheduler. It is enabled only when the window is in the *inactive* state.

When you select **Setup Scheduler**, NightSim carries out the following actions:

- Determines whether there is an existing scheduler that is associated with the same *key* as that entered in the current window.

If an active scheduler with the same scheduler key as the key you selected already exists on the target system to which you assigned processes, NightSim will ask you if you wish to abandon the setup operation or to replace the existing schedule with the new one. Creates a scheduler that is configured according to the parameters specified in the Scheduler Definition area of the window (see “Using the Scheduler Definition Area” on page 4-12 for details).

- Schedules the processes that are listed in the Processes Area of the window and starts them running (see “Using the Processes Area” on page 4-15 for details). Programs will run up to the first **fbwait** system call.
- Attaches the interrupt source to the scheduler.
- Enables the **Start** function. Initially, the simulation is stopped.
- Places the window in the *active* state and changes the state icon accordingly.

The **Setup Scheduler** menu option is equivalent to using both the  icon on the scheduler toolbar (see “The Toolbar” on page 3-12) and the **Set up** button in the Scheduler Control area of the window (see “Using the Scheduler Control Area” on page 4-9).

Start/Restart Scheduler


Mnemonic: S

This option starts or restarts a stopped simulation. It is enabled only when the simulation is stopped.

When you select **Start/Restart Scheduler**, NightSim carries out the following actions:

- If a real-time clock is being used as the interrupt source, sets the clock period in accordance with the value entered in the **Clock Period/Freq** field in the Scheduler Definition area (see “Clock Period/Freq” on page 4-15).

- Starts the simulation with the values of the *minor cycle*, *major frame*, and *overrun* counts set to zero.

The **Start/Restart Scheduler** option is equivalent to using both the  icon on the scheduler toolbar (see “The Toolbar” on page 3-12) and the **Start** button in the Scheduler Control area of the window (see “Using the Scheduler Control Area” on page 4-9).


Pause Scheduler

Mnemonic: P

This option stops the currently running simulation. It is enabled only when the simulation is running.

When you select **Pause Scheduler**, NightSim stops the simulation. When the simulation is stopped, the interrupt source is not detached from the scheduler, but the **Clock Period/Freq** field in the **Interrupt Definition** area are enabled. If the interrupt source is a real-time clock, you may wish to change the clock *period*.

While the simulation is stopped, you may also wish to change one or more of the parameters with which you have scheduled a process on the scheduler.


The **Pause Scheduler** option is equivalent to using both the  icon on the scheduler toolbar, which toggles between pause and resume, (see “The Toolbar” on page 3-12) and the **Pause** button in the Scheduler Control area of the window (see “Using the Scheduler Control Area” on page 4-9).

Resume Scheduler

Mnemonic: R

This option starts the stopped simulation from the point at which it was stopped. It is only enabled when the simulation has been running but is currently stopped.

When you select **Resume Scheduler**, NightSim restarts the simulation with the values of the *minor cycle*, *major frame*, and *overrun* counts the same as they were when you stopped the simulation.

The **Resume Scheduler** option is equivalent to using both the  icon on the scheduler toolbar, which toggles between pause and resume, (see “The Toolbar” on page 3-12) and the **Resume** button in the Scheduler Control area of the window (see “Using the Scheduler Control Area” on page 4-9).

Remove Scheduler

Mnemonic: R

This option allows you to remove the active scheduler from the system. It is enabled only when the window is in the *active* state.


When you select **Remove Scheduler**, NightSim carries out the following actions:

- Stops the simulation if it is running.

- If a real-time clock is being used as the interrupt source, and the scheduler is running, stops the clock.
- Detaches the interrupt source from the scheduler or disables *end-of-cycle scheduling*.
- Removes the scheduler from the system.
- Depending on your selected preference, NightSim will either kill the processes running on the scheduler, leave them running, or ask you which action you want to take.

The preference is set in the **Scheduler** area of the **Preferences Dialog** (see “Preferences Dialog” on page 3-14 for more information).

If the processes are left running, they may exit on their own, depending on how they are coded. A typical coding style is to loop as long as `fbwait()` returns a non-negative value. `fbwait()` will return `-1` on the next return from `fbwait()` after the scheduler has been removed (and `errno` will be set to `ENOENT`).

The Remove Scheduler option is equivalent to using both the  icon on the scheduler toolbar (see “The Toolbar” on page 3-12) and the **Remove** button in the Scheduler Control Area of the window (see “Using the Scheduler Control Area” on page 4-9).

Using the Scheduler Control Area

The Scheduler Control area allows you to create and remove a *frequency-based scheduler* and start, stop and resume a simulation with push buttons that you use to perform these functions. Status of the simulation run is also provided.

The Scheduler Control area is shown in Figure 4-3.

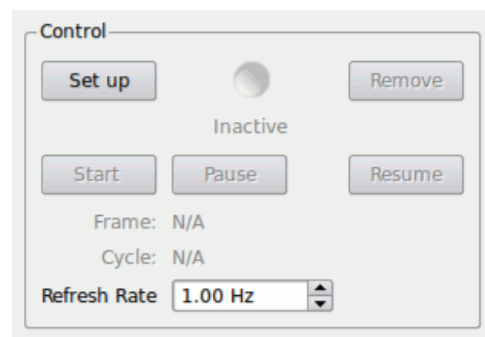



Figure 4-3. Scheduler Control Area

Descriptions of the buttons and fields on this page follow:


Set up Button

The **Set up** button is used to set up the configured scheduler. It is enabled only when the window is in the *inactive* state.

Pressing the **Set up** button is equivalent to using both the  icon on the scheduler toolbar (see “The Toolbar” on page 3-12) and the **Setup Scheduler** selection of the **Scheduler** menu. See **Setup Scheduler** on page 4-7 for a complete description of the operations that are performed.


Remove Button

The **Remove** button allows you to remove the active scheduler from the system. It is enabled only when the window is in the *active* state.

Pressing the **Remove** button is equivalent to using both the  icon on the scheduler toolbar (see “The Toolbar” on page 3-12) and the **Remove Scheduler** selection from the **Scheduler** menu. See **Remove Scheduler** on page 4-8 for a complete description of the operations that are performed.

Start Button


The **Start** button starts or restarts a stopped simulation. It is enabled only when the simulation is stopped.

Pressing the **Start** button is equivalent to using both the  icon on the scheduler toolbar (see “The Toolbar” on page 3-12) and the **Start/Restart Scheduler** selection in the **Scheduler** menu. See **Start/Restart Scheduler** on page 4-7 for a complete description of the operations that are performed.

If the simulation has been paused, the **Start** button label changes to **Start over** and when clicked, the simulation is restarted with all values starting at zero.

Pause Button


The **Pause** button stops the currently running simulation. It is enabled only when the simulation is running.

Pressing the **Pause** button is equivalent to using both the  icon on the scheduler toolbar, which toggles between pause and resume, (see “The Toolbar” on page 3-12) and the **Pause Scheduler** menu item in the **Scheduler** menu. See **Pause Scheduler** on page 4-8 for a complete description of the operations that are performed.

Resume Button

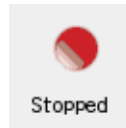
This option starts the stopped simulation from the point at which it was stopped. It is enabled only when the simulation has been running but is currently stopped.

When you select **Resume**, NightSim restarts the simulation with the values of the *minor cycle*, *major frame*, and *overrun* counts the same as they were when you stopped the simulation.

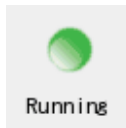
Pressing the **Resume** button is equivalent to using both the  icon on the scheduler toolbar, which toggles between pause and resume, (see “The Toolbar” on page 3-12) and the **Resume Scheduler** menu item in the **Scheduler** menu. See **Resume Scheduler** on page 4-8 for a complete description of the operations that are performed.

State of the Simulation

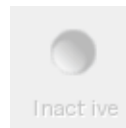
An icon is provided on the Scheduler Control area that indicates whether the simulation is stopped, running or inactive.



When the simulation is stopped, NightSim displays a red orb.



When the simulation is running, NightSim displays a green orb.



When the window is inactive, the state of the simulation area is grayed.

Frame

This field shows the number of the current major frame for the simulation running on the scheduler.

Cycle

This field shows the number of the current minor cycle for the simulation running on the scheduler.

Refresh Rate

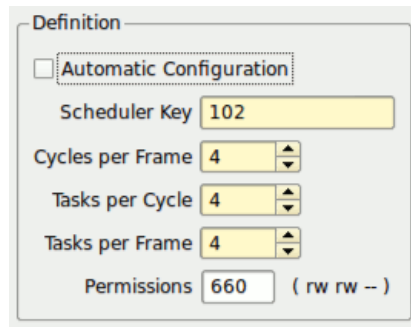
This field allows you to specify the frequency with which the frame and cycle counts are to be updated. Enter the number of seconds to occur between updates. The minimum value that you may specify is 0.01 second; the maximum is 100 seconds. The default value is 1.0 second.

This value also controls the frequency of Performance Monitoring updates on the Monitor page.

Using the Scheduler Definition Area

The Scheduler Definition area allows you to configure a *frequency-based scheduler*, specify maximum values for the scheduler, and specify the permissions that are to be associated with it.

The Scheduler Definition area is shown in Figure 4-4.



The image shows a configuration window titled "Definition". It contains several fields: a checkbox for "Automatic Configuration" which is unchecked; a text field for "Scheduler Key" containing the value "102"; three spinners for "Cycles per Frame", "Tasks per Cycle", and "Tasks per Frame", all set to the value "4"; and a text field for "Permissions" containing the value "660" followed by the octal permissions "(rw rw --)".

Figure 4-4. Scheduler Definition Area

Descriptions of the fields contained in the Scheduler Definition area follow:

Automatic Configuration

If this field is checked, NightSim chooses reasonable scheduler dimensions based on the start cycle and period values of the processes scheduled. A frame size is chosen such that each process will be scheduled at least twice within a frame.

If this field is unchecked, the user is responsible for choosing appropriate scheduler dimensions.

As a convenience, NightSim chooses scheduler dimensions with enough extra space to allow one */idle* process per CPU and at least one additional process to be added after the scheduler is set up.

Scheduler Key

This field allows you to specify a *key* for the frequency-based scheduler that you wish to create. The key is a user-chosen numeric identifier with which the scheduler will be associated. It can be any positive integer value. Note that the number of schedulers that can be configured at one time cannot exceed the value of FBSMNI, which is the maximum number of frequency-based schedulers permitted on your system (see "Establishing the NightSim Environment" on page 2-1 for configuration information).

Cycles per Frame

This field allows you to specify the number of cycles that compose a frame on the specified scheduler.

Tasks per Cycle

This field allows you to specify the maximum number of processes that can be scheduled to execute during one cycle.

Tasks per Frame

This field allows you to specify the number of scheduled tasks that may be initiated in a single frame of **Cycles per Frame**. Every time a process (or **/idle**) is scheduled to run within a frame constitutes a single task.

Permissions

This field allows you to specify the permissions required for operations related to the specified scheduler. Enter three octal digits—the first indicating permissions granted to the owner, the second indicating those granted to the group, and the third indicating those granted to other users. The octal method for setting permissions associated with a scheduler is the same as that used for specifying *mode* with the **chmod** command (for assistance in using this method, see the system manual page for **chmod(1)**). The default, **600**, grants read and alter (write) permission to the owner only.

Using the Scheduler Interrupt Area

The Scheduler Interrupt area allows you to select the interrupt source that is to be used for the scheduler.

The Scheduler Interrupt area is shown in Figure 4-5.

The screenshot shows a configuration window titled "Interrupt". It contains the following fields and controls:

- Interrupt Source:** A dropdown menu currently showing "Real-time clock rtc2".
- Interrupt Device:** A text field containing the path "/dev/rcim/rtc2".
- Clock Period/Freq:** A text input field containing the value "10.0", followed by a dropdown menu set to "mSec".
- Valid Range:** Text below the input field indicating the range "(0.001 - 655350.0) mSec".

Figure 4-5. Scheduler Interrupt Area

Descriptions of the fields contained in the Scheduler Interrupt area follow:

Interrupt Source

This field allows you to select the interrupt source for the scheduler. The interrupt source can be the end of a *minor cycle*, a real-time clock, an edge-triggered interrupt, or a user-supplied device. Figure 4-6 presents an example of the dropdown menu.

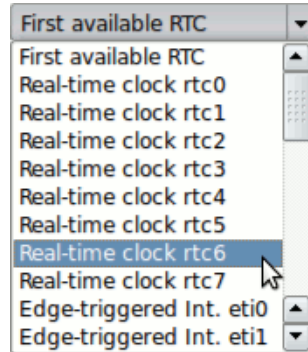


Figure 4-6. Interrupt Source Dropdown Menu

This list contains the set of devices available on the target system. Note that the options that NightSim displays vary according to the devices that are configured on the target system and may change if the target system is changed.

If you select the **End of processing cycle** option from the **Interrupt Source** dropdown menu, scheduling is triggered when the last process that is scheduled during the current *minor cycle* of the current *major frame* completes its processing.

If you select a real-time clock, the **Clock Period/Freq** text field is enabled so that you can specify the information needed to set the clock.

If you select the **Other...** option, NightSim displays the a simple text dialog which allows you to specify the name of a user-supplied device that you wish to use as the interrupt source. The **Device Pathname** must be the name of a device file that is supported by the RedHawk kernel as a FBS timing device.

If NightSim detects the device can be programmed as a real-time clock, it allows you to enter the cycle frequency. Otherwise, if it detects the device is a distributed interrupt source or edge triggered interrupt, it will arm and enable the device accordingly (when the scheduler is **Setup**). NightSim make these determinations by opening the device an issuing `ioctl()` calls as described in the *Realtime Clock and Interrupt Module (RCIM)* manual.

Interrupt Device

This area displays the actual interrupt device path used when the interrupt source is something other than **End Of Cycle Processing**. The device pathname is not always filled in until the scheduler is set up.

Clock Period/Freq

If you select a real-time clock as the interrupt source for the scheduler, this field allows you to specify a value to be used in setting the clock. Otherwise, this field is disabled.

You may specify the period in microseconds, milliseconds, hertz, or kilohertz by clicking on the dropdown and choosing the desired option. Figure 4-7 presents the options from which you may choose.

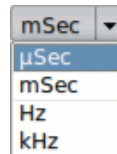


Figure 4-7. Clock Period/Freq Dropdown

Valid Range

The Valid Range field shows the minimum and maximum values that are permitted for the clock period specified in the Clock Period/Freq field. The range changes according to the option you choose.

The valid range of the clock period is determined by multiplying the minimum and maximum clock counts, 1 and 65535, by the clock resolution of 1, 10, 100, 1000, or 10000 microseconds.

Using the Processes Area

The Processes area allows you to:

- add programs to a *frequency-based scheduler*
- remove *FBS*-scheduled processes from a scheduler
- reschedule *FBS*-scheduled processes
- view scheduling data for current *FBS*-scheduled processes

PID ▲	Program	CPU IDs	Policy	Prio	Param	Limit	Ov Halt	Deadline	DI Halt	Origin	Schedule			
3100	steady	0-3	FIFO	22	0	1	false	--	--	--	0	1	2	3
2996	radar	2	FIFO	10	105	1	false	4.0 (msec)	false	Cycle	0	1	2	3
2997	weapons	0-3	RR	50	208	1	false	--	--	--	0	1	2	3

Key: Idle Scheduled Deadline Unscheduled

Refresh Add Copy Edit Remove

Figure 4-8. Processes Area

The Processes area contains a scrolling list of the processes that are currently scheduled on a *frequency-based scheduler* and the parameters with which they have been scheduled (CPU bias, scheduling policy and priority, starting *minor cycle*, and so on). You can select the process or processes that you wish to reschedule or delete by clicking on the respective item(s) in this list.

In addition, the Processes area includes a graphical representation of how the program will be scheduled within each frame. The labels show the number of the *minor cycle*. The colors indicate the current condition of the process: idle, scheduled, scheduled with a deadline or unscheduled as represented in the **Key**.

Double clicking on certain parameter fields in the Processes area displays the **Edit Process** dialog that allows you to modify the values in those fields. See the section “Edit Process” on page 4-21 for a description of the **Edit Process** dialog.

Columns in the Processes area can be resized or hidden using selections from the context menu. The context menu is displayed by right-clicking in the Processes area or on a specific column heading.

The Processes area contains a row of process control buttons below the list of FBS-scheduled processes. These controls (with the exception of **Refresh**) are also provided by the Process Menu of the NightSim window menu bar and by a context menu that is accessible by right-clicking within the Process area. These controls are explained under the section “The Process Menu” on page 4-16. The **Refresh** button refreshes the Processes area.

The Process Menu

Processes to be scheduled on a frequency-based scheduler can be selected using the buttons at the bottom of the Processes area and from the context menu that is accessed by right-clicking within the Process area.

The following figure illustrates NightSim’s **Process** menu.

Resiz <u>e</u> Columns to Data	
Resiz <u>e</u> Columns to <u>D</u> efault width	
<u>A</u> dd Process	Ctrl+N
<u>C</u> opy Process	Ctrl+C
<u>E</u> dit Process	Ctrl+E
<u>R</u> emove Process(es)	
Debug <u>R</u> unning Process	
Select Custom <u>F</u> ields...	
<u>H</u> ide this Field	

Figure 4-9. Process Menu

The following paragraphs describe these options in more detail.

Add

Mnemonic: A
Accelerator: Ctrl+N

This selection allows you to add a program to a frequency-based scheduler or measure idle time on a CPU or reschedule an FBS-scheduled process. When you make this selection, NightSim displays the **Add Process** dialog. Note that if you select a process from the list of scheduled processes and then select Add, the dialog contains the scheduling parameters for the selected process. See the section “Add Process” on page 4-19 for a description of the Add Process dialog.

Copy

Mnemonic: C
Accelerator: Ctrl+C

This selection adds a copy of one or more selected process(es) to the frequency-based scheduler. When you click on a process in the Processes area, this selection is activated.

Edit

Mnemonic: E
Accelerator: Ctrl+E

This selection allows you to edit attributes of FBS-scheduled processes on a scheduler. When you click on a process in the Processes area, this selection is activated. When you make this selection, NightSim displays the **Edit Process** dialog. The dialog contains the scheduling parameters for the selected process. See the section “Edit Process” on page 4-21 for a description of the Edit Process dialog.

Remove

Mnemonic: R

This selection allows you to remove FBS-scheduled processes from a scheduler. All processes that are selected in the list will be removed.

If the scheduler is active when you select **Remove**, NightSim displays a dialog to confirm killing the selected processes. You may press the **Yes** button to kill the selected processes after deleting them from the *frequency-based scheduler*. You may also press the **No** button to delete the processes from the scheduler but allow them to continue executing or press **Cancel** to abort the action.

If the scheduler is inactive when you select **Remove**, NightSim displays a dialog to confirm deleting the processes. You may press **OK** to delete the processes from the scheduler or **Cancel** to cancel the operation.

Debug Running Process

Mnemonic: G

This selection invokes the NightView debugger and attaches to the selected process. Once the debugger finishes the attach step, the process is automatically resumed. You can debug it without stopping it (thereby not affecting its timing), or stop it and debug traditionally.

Note that depending on the length of your scheduler's cycle, the act of attaching to the process may cause it to overrun during attach processing.

Select Custom Fields...

Mnemonic: F

This option allows you to specify the columns that are to be displayed in the Processes area.

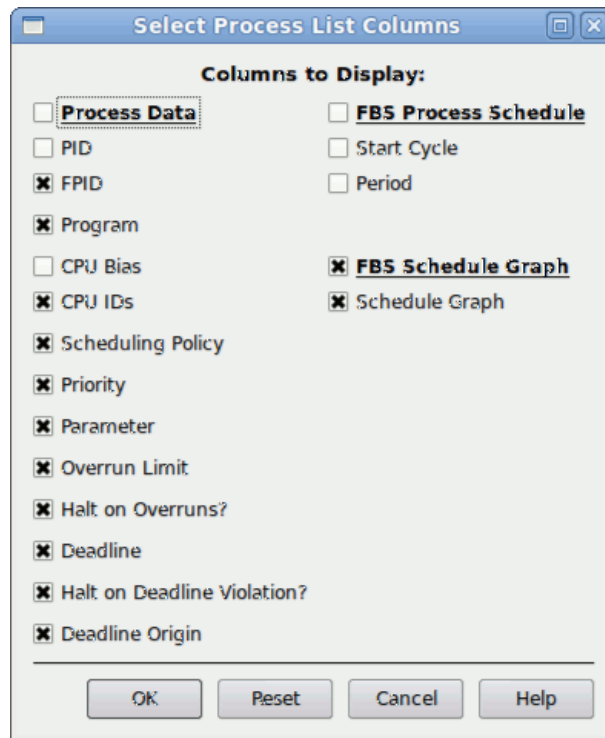


Figure 4-10. Process Columns to Display Dialog

You use the panels of check boxes to select the types of values that you wish to save. Click on one or more boxes as desired.

After you have selected the types of values that you wish to save, select OK to apply the selections and close the dialog box, or you can reset the original settings, cancel the selections, or display help related to the dialog.

Add Process

When the Add option in the Processes area is clicked, the Add Process dialog appears.

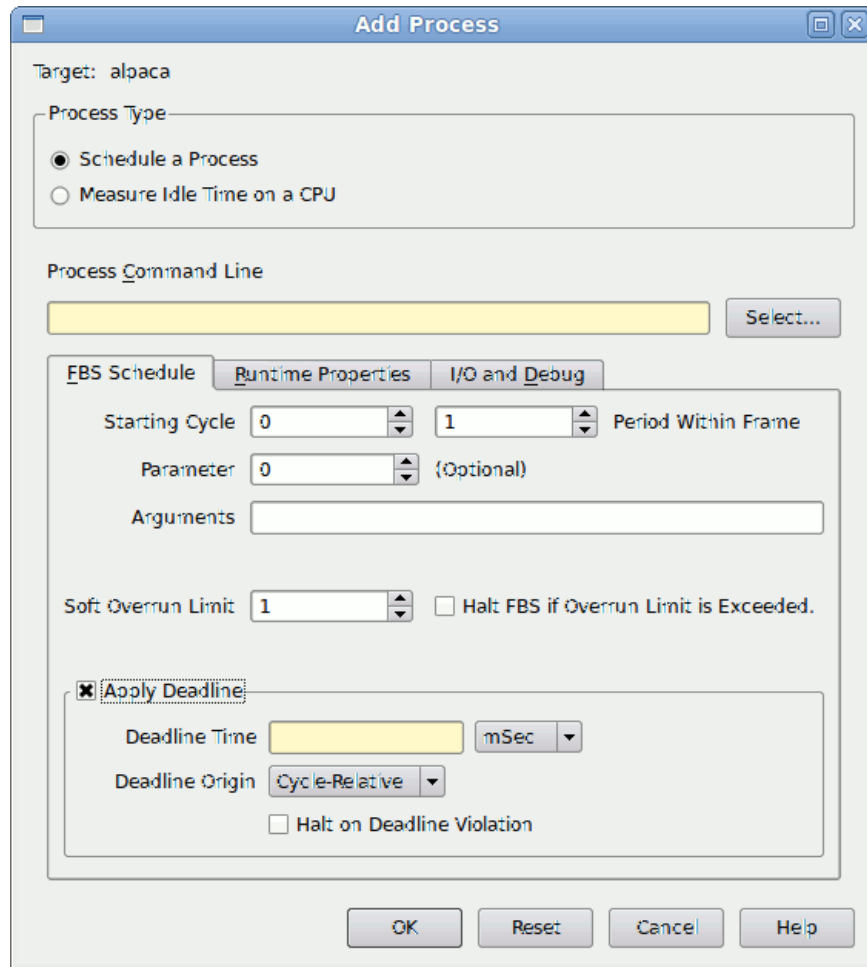


Figure 4-11. Add Process Dialog

The following fields appear at the top of the Add Process dialog:

Schedule a Process

When this radio button is filled, the dialog provides information for scheduling and defining a process to be added to the scheduler.

Measure Idle Time on a CPU

When this radio button is filled, the dialog shows the CPU bias of the idle task. When the Add option is selected, the `/idle` task is added to the scheduler. See “Monitoring Idle Time” on page 5-6 for more information about monitoring idle time.

Command Line

You may specify the path to the program to be scheduled on the scheduler. Arguments are accepted for all programs except **/idle**. The path must be a valid path name on the *target system*.

Select...

You may use the **Select** button to bring up the **Select a program name** dialog to specify the directory containing the program you want to configure under the *FBS*. This button works on the path of the program, leaving arguments intact.

The selection dialog operates on the host system where NightSim was invoked. If the target system is remote, the file path you choose from this dialog may not be valid on the target system. The program file does not need to exist on the host system if the target is remote.

The tabs on this dialog are used to display different pages that contain various parameters related to the given process. Refer to the section “Add Process and Edit Process Dialog Pages” on page 4-22 for complete descriptions.

The buttons along the bottom of the dialog control actions to be taken after you have entered the scheduling parameters for a specified program:

OK

Clicking on the **OK** button makes the specified changes, if any, and closes the dialog.

Reset

Clicking on the **Reset** button resets the fields to the original values without closing the dialog.

Cancel

Clicking on the **Cancel** button resets the fields to the original values if changed and closes the dialog.

Help

Clicking on the **Help** button displays help for the dialog.

Edit Process

The **Edit Process** dialog is identical to the Add Process dialog, except that you cannot change the Process type.

Add Process and Edit Process Dialog Pages

The bottom portion of the Add Process and Edit Process dialogs is divided into a number of pages that contain various parameters related to the given process. These pages are:

- FBS Schedule (see “FBS Schedule” on page 4-22)
- Runtime Properties (see “Runtime Properties” on page 4-25)
- I/O and Debug (see “I/O and Debug” on page 4-28)

FBS Schedule

The FBS Schedule page of the Add Process and Edit Process dialogs (see “Add Process and Edit Process Dialog Pages” on page 4-22) allows the user to specify parameters such as the starting cycle, period, and overrun limits for the given process.

The FBS Schedule page of the Edit Process dialog is shown in Figure 4-12:

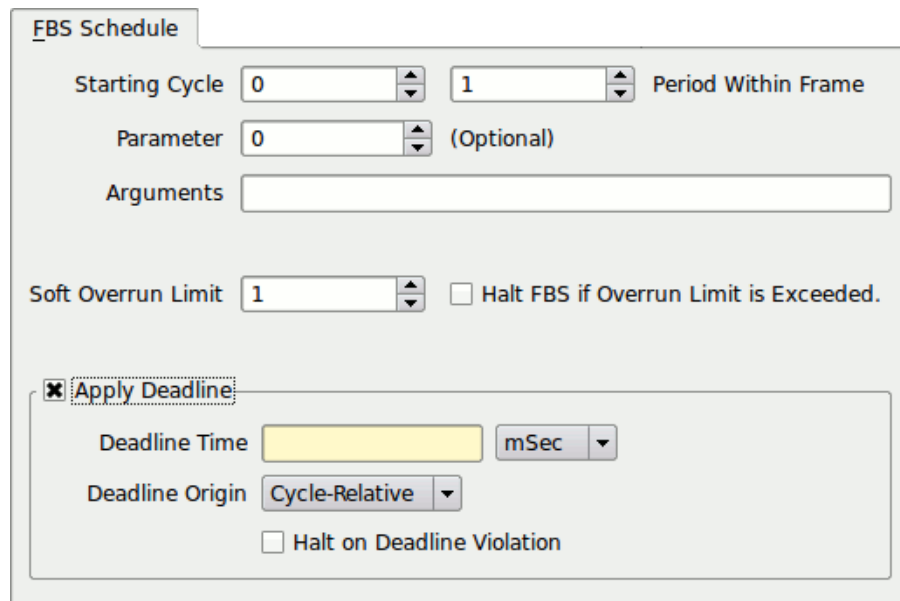


Figure 4-12. Add Process and Edit Process Dialogs - FBS Schedule Page

Descriptions of the fields on the FBS Schedule page follow:

Starting Cycle

This field allows you to specify the first *minor cycle* in which the specified program is to be wakened in each *major frame*. Enter a number ranging from zero to the total number of minor cycles per frame minus one. (The total number of minor cycles per

frame is defined when you use the Scheduler Definition Area to configure a scheduler. See “Using the Scheduler Definition Area” on page 4-12 for details.)

NOTE

You must enter a starting cycle of zero for unscheduled processes (for an explanation of unscheduled processes, see “Using the Monitor Area” on page 5-1 and the section entitled “Monitoring Unscheduled Processes” on page 5-6).

Period Within Frame

This field allows you to establish the frequency with which the specified program is to be wakened in each *major frame*. A period of one indicates that the specified program is to be wakened every *minor cycle*; a period of two indicates that it is to be wakened once every two minor cycles, a period of three once every three minor cycles, and so on. Enter the number of minor cycles representing the frequency with which you wish the program to be wakened. This number can range from zero to the number of minor cycles that compose a frame on the scheduler. (The total number of minor cycles per frame is defined when you use the Scheduler Definition Area to configure a scheduler. See “Using the Scheduler Definition Area” on page 4-12 for details.)

NOTE

You must enter a period of zero for unscheduled processes (for an explanation of unscheduled processes, see the section entitled “Monitoring Unscheduled Processes” on page 5-6).

Parameter

This field allows you to pass an integer value to a process that is scheduled on a frequency-based scheduler. The value must be a 32-bit decimal value. The default value is 0. The process can retrieve this value by using the `sched_pgmqry(3)` library routine. If no other arguments are specified (see the **Arguments** field), then this value is also passed (as an ASCII string) as the first argument to the process which can be retrieved through conventional means (e.g. `argv[1]`).

Arguments

This field allows you to pass arbitrary arguments to the process. The arguments specified here can be retrieved through conventional means in your program. For example, a C or C++ program would retrieve the arguments by referencing the variables `argv[1]` through `argv[argc-1]` in the `main` procedure. You may quote or escape characters using conventional shell rules so that specific white space does

act as an argument separator. For example, if the **Arguments** text field contained the exact characters from the next line,

```
How\ long\ is\ a piece "of string?"
```

then, three arguments would be passed to the process.

If you specify arguments here, the **Parameter** value will not be passed as an argument to your process, but it can be retrieved using **sched_pgmqry(3)**.

Soft Overrun Limit

This field allows you to specify a consecutive overrun limit count that this process will tolerate and have processed as soft overruns by the kernel. *Soft overruns* are catastrophic failures only if the process reaches its limit on the number of soft overruns tolerated. This soft overrun limit defaults to 0 for each process.

NOTE

If the **Period Within Frame** of a process is changed while the scheduler is running, the Monitor area will not reflect the updated value until the next Monitor area update after the process runs again.

Halt on Overrun

This check button allows you to indicate whether or not the selected scheduler should be stopped in the event that the specified program causes a *frame overrun*. Check this field if you wish the scheduler to be stopped.

NOTE

This may be useful in conjunction with scheduling a process under the NightView debugger, making it easier to stop a scheduled process at the source of an overrun and determine the activity which caused the overrun to occur.

Apply Deadline

The FBS permits you to apply a deadline to a scheduled process. If the process fails to return to **fbswait(3)** (i.e. complete its processing) before the deadline expires, a deadline violation occurs. The "Apply Deadline" group box will be active when the target system has support for deadlines, or when NightSim is in Off-Line mode. The following fields specify a deadline for the process:

Deadline Time

This input field accepts a floating point value denoting the duration of the deadline. The associated menu specifies the units for the deadline time value.

Deadlines represent real time, not just CPU time. Deadline times may be specified in microseconds, milliseconds, or seconds. In addition, NightSim allows you to specify a number of minor cycles as a deadline time, if the scheduler uses a Clock Interrupt Source.

NOTE

The “Apply Deadline” group box is only enabled when the chosen target system kernel supports the deadline feature, and when NightSim is in Offline Mode.

Deadlines are supported in RedHawk 5.2.1 and later versions of the Concurrent FBS product.

Deadline Origin

This menu allows you to select the start time from which deadlines are measured.

The default, Cycle-relative origin, begins timing the process from the beginning of the cycle in which the process is scheduled to run. Choose cycle-relative origin if it is important that the process completes its processing at a fixed time after the cycle begins, even if it is delayed by a higher priority process on the same CPU.

The Task-relative origin option begins timing the process when it returns from `fbwait()` and begins its processing. Choose a task-relative origin if it is important that the process complete its processing within a fixed amount of time once it begins processing, even if it is delayed by a higher priority process.

Halt on Deadline Violation

This check box causes the scheduler to be halted when a deadline violation is detected. Violations are detected when the process calls `fbwait()` or `sched_pgm_deadline_test()`.

Runtime Properties

The Runtime Properties page of the Add Process and Edit Process dialogs (see “Add Process and Edit Process Dialog Pages” on page 4-22) allows the user to specify the working directory, CPU bias, scheduling policy and priority of the process.

The Runtime Properties page of the Edit Process dialog is shown in Figure 4-13:

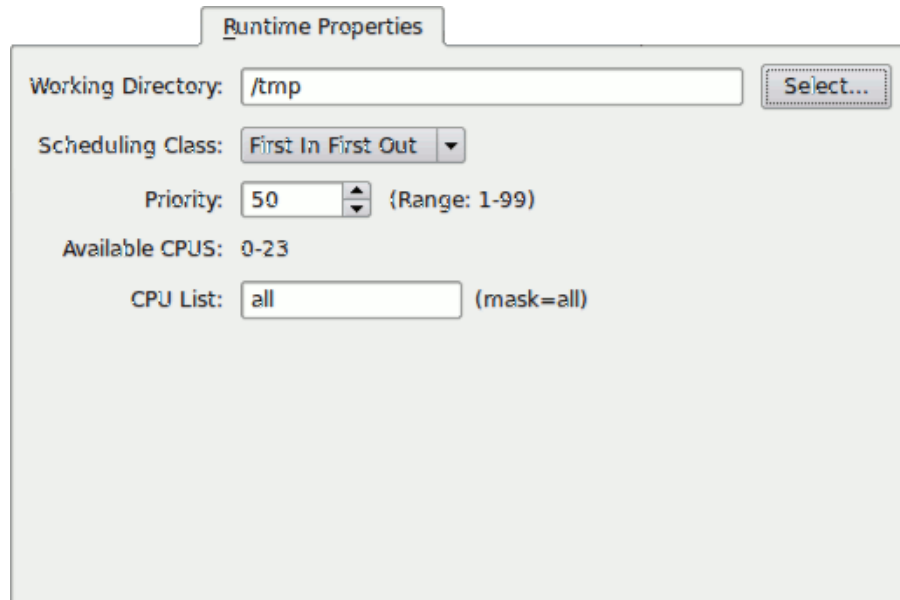


Figure 4-13. Add Process and Edit Process Dialog - Runtime Properties Page

Descriptions of the fields on the Runtime Properties page follow:

Working Directory

The Working Directory affects relative pathnames for the process to be scheduled (see “Command Line” on page 4-21), its working directory, and any files specified as standard input or standard output (see “Input File” on page 4-29 and “Output File” on page 4-30).

If this field is left blank, then the working directory used for all processes launched through NightSim will be either the current working directory from which NightSim was launched (if the target system is the local host system), or the user’s home directory on the target system (if the target system is a remote system).

If you specify a directory name here, remember that it must be accessible from the target system.

Select...

You may use the Select button to bring up the Select a working directory dialog to specify the working directory for the program you want to configure under the *FBS*. Note that the file selection dialog operates solely on the file systems of the host system where NightSim runs -- be sure to select a directory path which is accessible from the target system.

Policy

This panel of radio buttons allows you to select the POSIX scheduling policy for the specified program. The options are as follows: the **FIFO** (first-in-first-out) scheduling policy, the **Round Robin** scheduling policy, and the **Other** (interactive) scheduling policy.

Priority

The range of priority values that you can enter is governed by the scheduling policy specified. You can determine the allowable range of priorities associated with each policy (FIFO, Round-Robin, or Other) by invoking the **run(1)** command from the shell and not specifying any options or arguments (see the corresponding system manual page for an explanation of this command). Higher numerical values correspond to more favorable scheduling priorities.

NightSim displays the range of priority values that you can enter next to the Priority field.

NOTE

If the Priority of a process is changed while the scheduler is running, the Monitor Area will not reflect the updated value until the next Monitor Area update after the process runs again.

CPU List

Process execution will be constrained to the CPUs listed here.

By default, the list is `all`, which means the process can run on any CPU on the target system which isn't shielded from process execution (consult the **shield(1)** man page for more information on shielding).

The list can either be the word `all`, or a comma-separated list of CPU numbers or ranges of CPU numbers; for example: `0, 2-3`.

To the right of the text field a description of the resultant CPU mask is shown. Some system interfaces require CPU affinity to be specified as a mask, with each bit in the mask representing a CPU. The mask is shown to remind you that the numbers you enter into the text field here are logical CPU numbers, not hexadecimal characters in a CPU mask.

If you enter something invalid into the text field, the description to the right changes to the word `invalid`, shown in red. Ultimately, syntactically-invalid CPU lists are automatically replaced with a list indicating `all`.

NOTE

If the CPU List of a process is changed while the scheduler is running, the Monitor Area will not reflect the updated value until the next Monitor Area update after the process runs again.

NOTE

You cannot change the CPU List of `/idle` while the scheduler is active because it is not a real process. You must either add a new process with the desired CPU List, or make the change while the scheduler is inactive. See “How Is Idle Time Monitored?” on page 5-6 for more information.

I/O and Debug

The I/O and Debug page of the Add Process and Edit Process dialogs (see “Add Process and Edit Process Dialog Pages” on page 4-22) allows the user to specify where to read input from, where to send output, and if the process is to be scheduled within a Night-View dialog or within an X terminal window.

The I/O and Debug page of the Edit Process dialog is shown in Figure 4-14:

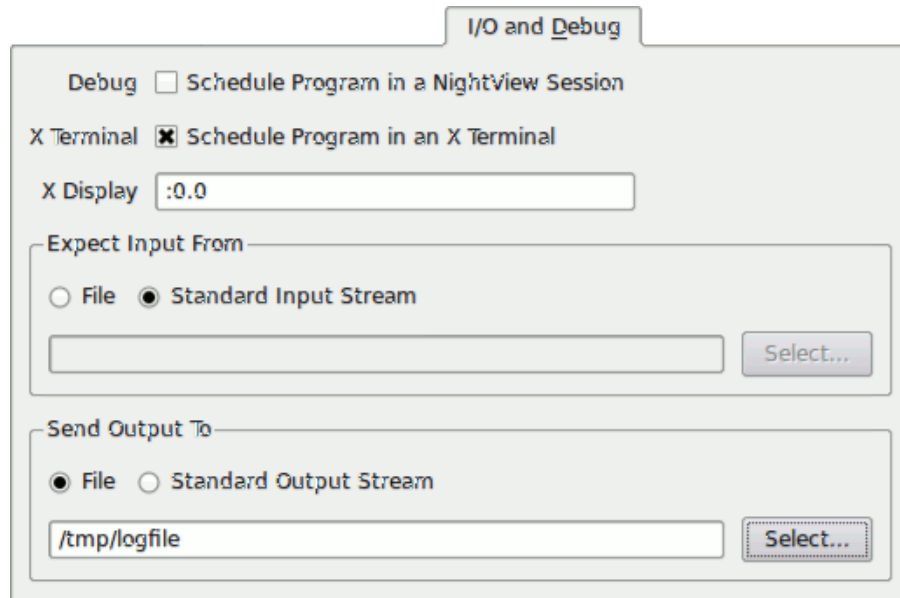


Figure 4-14. Add Process and Edit Process Dialog - I/O and Debug Page

Descriptions of the fields on the I/O and Debug page follow:

Debug

Schedule this process under the NightView debugger. When the **Schedule program in a NightView session** checkbox is checked, the process will be stopped at the point it begins execution, allowing early intervention for the purpose of setting breakpoints, watchpoints, etc. A **resume** command to NightView is required to resume the execution of the scheduled process.

X Terminal

Schedule the process in an X terminal window. By default, **gnome-terminal** is used as the X terminal window program. You can alter this behavior by setting your X terminal preference, see “Preferences Dialog” on page 3-14 for more information.

X Display

The X Display field may be used to provide a valid X display for use when scheduling the process in an X terminal window or if your process has graphical X requirements.

Normally you should leave this field blank. When blank, the display is automatically set by NightSim. If the target system is a remote system, the display used by the active **ssh** connection (ssh is used to communicate to the target system by NightSim) is used. If the target system is the local system, then the display used by NightSim is used by the process.

Expect Input From:

Select the input source, if any, for the scheduled process. By default, as in previous versions of NightSim, input is taken from the file, **/dev/null**. When either the **Debug** or **X Terminal** checkboxes are selected, the default is to permit input to come from the **Standard Input Stream**, but you may choose **File** input in either case if your scheduled program needs a fixed input stream (see **Input File** below).

Input File

If **File** is selected for **Expect Input From**, this field specifies the name of the input file which the scheduled process will use. By default, the file is **/dev/null**. See “Pathname Interpretation” on page 4-30.

The path must represent a valid readable file at the time the program scheduler is created (i.e. when the NightSim Scheduler transitions to an **Online** mode of operation).

Select...

You may use the **Select** button to bring up the **Select an Input File** dialog to specify the name of the input file which the scheduled process will use.

The selection dialog operates on the host system; if the target system is remote, the selected file path may not be valid on the target system.

Send Output To:

Output may be sent to an actual file specified in the **Output File** field, if desired. By default, output is sent to the **Standard Output Stream** of the scheduled process.

Output File

If **File** is selected for **Send Output To**, this field specifies the name of the output file to generate. The output file name may be a full pathname or may be a pathname relative to the **Working Directory** specified for the scheduled process. See “Pathname Interpretation” on page 4-30.

Select...

You may use the **Select** button to bring up the **Select an Output File** dialog to specify the name of the output file that will be generated.

The selection dialog operates on the host system; if the target system is remote, the selected file path may not be valid on the target system.

Pathname Interpretation

When entering a process or file name in the **Add Process** or **Edit Process** dialog, the user can make use of the following conventions:

- if the pathname begins with /, it is assumed to be fully rooted
- otherwise, it is considered to be relative to the path in the **Working Directory** field. If the **Working Directory** field is left blank, then the file name is relative to the current working directory from which NightSim was launched or the home directory of the user on the remote target system, respectively, depending on whether the target system is the local host system or a remote system.

Using the Monitor Area

This chapter describes the *performance monitor* and explains how to monitor *FBS*-scheduled processes, a processor's *idle time*, and unscheduled processes. It introduces the components of the Monitor area of the NightSim window and describes the use of each component.

Understanding the Performance Monitor

The performance monitor is a mechanism that enables you to monitor *FBS*-scheduled processes' utilization of a CPU.

The performance monitor provides you with the ability to:

- Obtain performance monitor values by process
- Log performance monitor values to a text file
- Start and stop performance monitoring
- Clear performance monitor values

All process times related to performance monitoring are CPU times. These times do not account for periods of time where the process is not executing on a CPU. Thus even though a process' CPU time may not exceed its scheduling constraints, the process may still overrun or miss its deadlines.

NOTE:

Deadlines are values of real-time, not just CPU time. A process misses its deadline if it does not reach the next call to *FBS* wait before the deadline has passed (real time). The performance monitor does not report on such times per se, but it does list the number of deadline misses.

You also have the ability to set the *timing mode* under which the performance monitor is to run. You can select one of two modes: one that includes time spent servicing interrupts in performance monitor timing values and one that excludes time spent servicing interrupts from those values.

When the performance monitor timing mode is set to include interrupt time, a process's user and system times will total the elapsed time that accrues when the process is the currently running process. This elapsed time includes time spent servicing interrupts and

performing context switches. Time spent servicing interrupts is added to the process's system time. Time spent switching to a new process is included in the new process's system time.

When the performance monitor timing mode is set to exclude interrupt time, a process's user and system times will total the time that accrues when the process is the currently running process. This time excludes time spent servicing interrupts, but it includes time spent performing context switches. Time spent switching to a new process is included in the new process's system time.

The timing mode (including or excluding interrupt time) is a system-wide setting and therefore applies to all FBS schedulers active on the system.

What Values Are Monitored?

The *performance monitor* keeps track of the CPU time that a process spends running from the time that it is wakened by a frequency-based scheduler until it calls `fbswait`, excluding time during which the scheduler was stopped. Time is calculated from the frame and cycle count and the clock period at the time the sample performance monitor data is taken, measured in microseconds. One instance of a process's being wakened by a scheduler is referred to as an iteration or a cycle. Performance monitor values for *FBS*-scheduled processes are reported both in terms of cycles, or *iterations*, and in terms of *major frames*. They reflect what has happened since the last time that performance monitor values were cleared and performance monitoring was enabled.

When performance monitoring is enabled for an *FBS*-scheduled process, NightSim maintains the following values for each process:

Process Parameters

Target System

The system on which this process is running

PID

The process's process identification number

FPID

The process's unique *frequency-based scheduler* identifier

Program Name

The process's name

Policy

The process's scheduling policy

Priority

The process's scheduling priority

Start Cycle

The number of the first *minor cycle* in which the process is to be wakened in each frame

Period

The frequency with which the process is to be wakened in each *major frame*

CPU Bias

The process's CPU mask

CPU Ids

A list of the CPU IDs enabled in the CPU mask

General Statistics

Parameter

An interval value passed to the process when it begins execution

Total iterations, cycles

The number of times that the process has been wakened by the scheduler

Last time

The amount of time that the process has spent running from the last time that it has been wakened by the scheduler until it has called **fbwait**

Total time

The total amount of time that the process has spent running in all cycles

Average time

The average amount of time that the process has spent running in all cycles, or *iterations*. This value is obtained by dividing the value reported in the Total Time column by the value reported in the Iterations column.

Soft Overruns

The number of times that the process has caused consecutive “soft” frame overruns.

Overruns

The number of times that the process has caused consecutive “hard” frame overruns

Overrun Limit

The number of soft overruns allowed before halting the scheduler

Halt on Overruns?

Whether to halt the scheduler when the soft overrun limit is reached

Percent of Period Used

This value is maintained only if the interrupt source for the *frequency-based scheduler* on which the process is scheduled is a real-time clock. It is obtained by dividing the value reported in the Average Time column by the amount of time that is available to the process. The amount of time available to the process is calculated from its *period* and the length of a clock cycle.

Deadline Violations

This value is maintained only if deadlines are supported on the target and the process had a deadline applied. It indicates the number of times that the process has missed an applied deadline. Unlike overruns, a deadline can be specified in any unit of time and is not therefore necessarily related to cycle periods. See “Apply Deadline” on page 4-24 for more information.

Minimum Values

Minimum cycle time

The least amount of time that the process has spent running in a cycle

Minimum cycle frame

The number of the *major frame* in which the *minimum cycle time* has occurred

Minimum cycle cycle

The number of the *minor cycle* in which the *minimum cycle time* has occurred

Minimum frame time

The least amount of time that the process has spent running during a *major frame*

Minimum frame frame

The number of the *major frame* in which the *minimum frame time* has occurred

Maximum Values

Maximum cycle time

The greatest amount of time that the process has spent running in a cycle

Maximum cycle frame

The number of the *major frame* in which the *maximum cycle time* has occurred

Maximum cycle cycle

The number of the *minor cycle* in which the *maximum cycle time* has occurred

Maximum frame time

The greatest amount of time that the process has spent running during a *major frame*

Maximum frame frame

The number of the *major frame* in which the *maximum frame time* has occurred

Monitoring Idle Time

The *performance monitor* provides you with the capability of monitoring a processor's idle time. Idle time refers to the time that the CPU is not busy. By monitoring a processor's idle time, you can determine the amount of CPU time that is available to be allocated to additional processes.

Procedures for monitoring idle time are described in "How Is Idle Time Monitored?" on page 5-6.

How Is Idle Time Monitored?

You can monitor a particular processor's *idle time* if you add the process `/idle` to a frequency-based scheduler and schedule it on the desired processor. You can also monitor idle time for a number of different processors by adding `/idle` to a selected *frequency-based scheduler* with multiple CPUs selected. One `/idle` process will be added per CPU selected.

NOTE

You can schedule `/idle` on a particular processor for a particular target only once. This is a system-wide restriction. Only one user per processor per target may schedule `/idle`.

To add `/idle` to a frequency-based scheduler, use the Processes Area of the NightSim window (see "Using the Processes Area" on page 4-15 for details). For an explanation of the procedures for adding a program to a scheduler, refer to "Add Process" on page 4-19

When you use NightSim to add `/idle` to a frequency-based scheduler, the only parameter that you may specify is the CPU bias mask. The CPU mask for `/idle` may specify one or more CPUs. If multiple CPUs are selected, `/idle` will be scheduled on each selected CPU. The default scheduling priority is zero. The starting base cycle is zero, and the period is one. `/idle` will be scheduled every *minor cycle*, starting with the first minor cycle in each *major frame*.

You can view scheduling information for `/idle` in the same way that you view it for other *FBS*-scheduled processes—in the Processes Area of the NightSim window (see "Using the Processes Area" on page 4-15 for details).

Monitoring Unscheduled Processes

The *performance monitor* provides you with the additional capability of monitoring the performance of unscheduled processes. Unscheduled processes are those that are not wakened by the scheduler and do not call `fbswait`; they are not scheduled to run at a certain frequency. To be able to obtain *performance monitor* values for such processes, you must first add them to a *frequency-based scheduler* and specify a starting base cycle of zero and a *period* of zero. The other scheduling parameters that you must specify include the process's scheduling priority and the CPU on which it is to be scheduled. You

can optionally specify a decimal value to be passed to a process that is scheduled on a frequency-based scheduler. The **Halt on overrun** flag does not apply to an unscheduled process.

You can add unscheduled processes to a frequency-based scheduler by using the Processes Area (see “Using the Processes Area” on page 4-15 for details). For an explanation of the procedures for adding a program to a scheduler, refer to “Add Process” on page 4-19.

You can view scheduling information for unscheduled processes in the same way that you view it for other *FBS*-scheduled processes—in the Processes Area (see “Using the Processes Area” on page 4-15 for details).

The *performance monitor* values that are maintained for unscheduled processes include the following:

- Last Time
- Total Time
- Min Frame
- Min Frame Time
- Max Frame
- Max Frame Time

You can obtain these values if you enable performance monitoring for the processes. Procedures for enabling performance monitoring and retrieving *performance monitor* values are described in detail in “The Monitor Menu” on page 5-9.

The Monitor Page

The Monitor page allows you to start and stop performance monitoring on *FBS*-scheduled processes, clear *performance monitor* values, and display *performance monitor* values for selected *FBS*-scheduled processes.

The Monitor page appears by clicking on the Monitor tab at the top of the screen or by selecting Monitor Page from the View menu or by using the Alt+O keyboard shortcut.

Figure 5-1 is an example of the Monitor page.

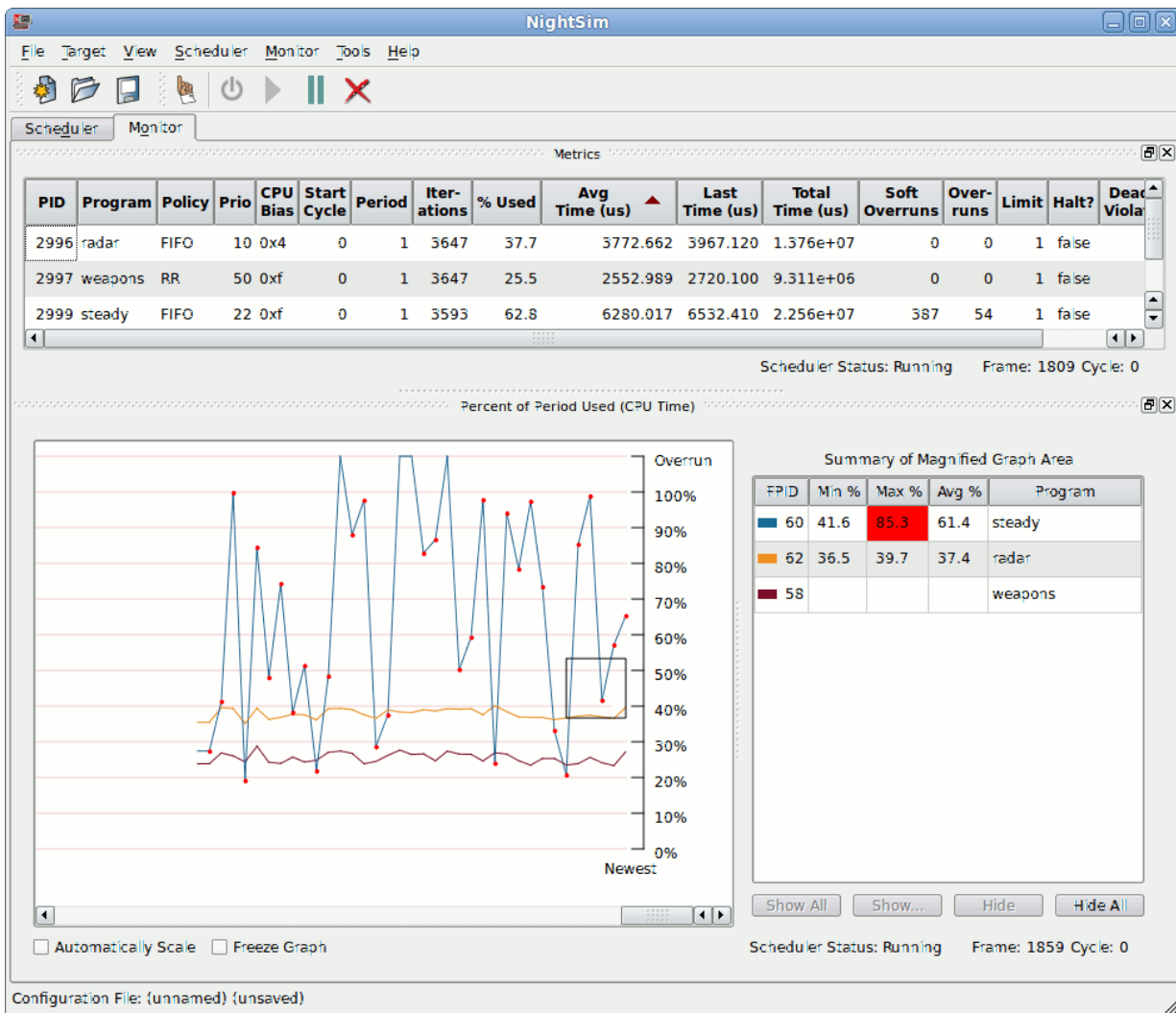


Figure 5-1. Monitor Page

The Monitor page consists of two panels: the Metrics panel and the Percent of Period Used panel.

The **Metrics** panel is a single table with each row representing a single process (or /idle). It contains attributes of the process at the time of the last periodic sample.

The **Percent of Period Used** panel provides a graph of CPU Usage for each process with points plotted over time. Each point in the graph represents the percentage of CPU time used by that process during one of its cycles. By default, there is one line for each process on the scheduler.

The **Monitor** page also allows you to specify which panels are to be displayed in the Monitor area and whether logging of data should occur (to a text file).

The Monitor Menu

The **Monitor** menu provides options for disabling monitoring, setting the timing mode, changing the display resolution and precision, selecting values, logging output to a file, and clearing data.

The **Monitor** menu appears on the NightSim window menu bar (see “The Menu Bar” on page 3-3). Figure 5-2 presents this menu.

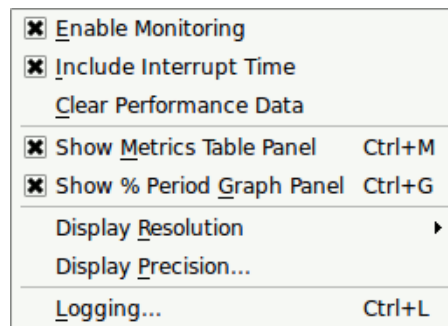


Figure 5-2. Monitor Menu

Descriptions of the options on the **Monitor** menu follow.

Enable Monitoring

Mnemonic: E

This item indicates the state of performance monitoring. By default, monitoring is enabled for all FBS-scheduled processes and this box is checked. To disable Performance Monitoring for all processes, uncheck this box.

Include Interrupt Time

Mnemonic: I

This item allows you set the *timing mode* under which the performance monitor is to run, indicating that time spent processing interrupts is to be included or excluded in

performance monitor timing values. This is a system-wide setting that affects all processes on the *target system* that have processes attached to any scheduler.

Clear Performance Data

Mnemonic: C

This option clears all *performance monitor* values for all processes on the scheduler. The Metrics table and Percent of Period Used graph are cleared as well. Performance data will begin to accumulate again immediately afterwards.

Show Metrics Table Panel

Mnemonic: M

Accelerator: Ctrl+M

This checkbox controls whether the Metrics panel is shown on the Monitor page.

Show % Period Graph Panel

Mnemonic: G

Accelerator: Ctrl+G

This checkbox controls whether the Percent of Period Used graph is shown on the Monitor page.

Display Resolution

Mnemonic: R

This option allows you to control the precision with which NightSim reports the *performance monitor* timing values. When you select this option, NightSim displays the menu that Figure 5-3 presents.

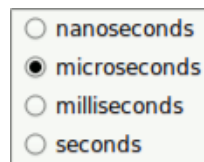


Figure 5-3. Resolution Options

You may specify the precision in nanoseconds, microseconds, milliseconds, or seconds by clicking on a radio button.

Display Precision...

This item allows you to control the number of fractional decimal digits used to display performance values associated with time. It affects minimum, maximum, average, and last cycle times. It does not affect cumulative time values.

Logging...

Mnemonic: L
 Accelerator: Ctrl+L

This item launches the Logging Dialog which allows you to enable or disable logging and select the filename and performance monitoring values to be logged. See “Logging” on page 5-21 for more information.

The Metrics Panel

The Metrics panel consists of a table and a context menu which has controls similar in nature to the Monitor menu discussed above.

PID	Program	Policy	Prio	CPU Bias	Start Cycle	Period	Iterations	% Used	Avg Time (us)	Last Time (us)	Total Time (us)	Soft Overruns	Overruns	Limit	Halt?
2996	radar	FIFO	10	0x4	0	1	10647	37.7	3774.124	3748.180	4.018e+07	0	0	1	false
2997	weapons	RR	50	0xf	0	1	10647	25.6	2558.467	2591.320	2.724e+07	0	0	1	false
2999	steady	FIFO	22	0xf	0	1	10515	62.5	6247.180	7007.460	6.569e+07	1080	132	1	false

Scheduler Status: Running Frame: 5309 Cycle: 0

Figure 5-4. The Metrics Panel

You may specify the types of information that are to be reported and the resolution with which performance monitor timing values are displayed (for example, microseconds, milliseconds, etc.).

NOTE

If the Priority, Period, or CPU Bias mask of a process are changed while the scheduler is running, the Monitor area will not reflect the updated values until the next Monitor area update after the process runs again.

Columns in the table can be resized or hidden using selections from the context menu. The context menu is displayed by right-clicking in the Monitor table or on a specific cell in the table.

The context menu provides you control over the table contents.

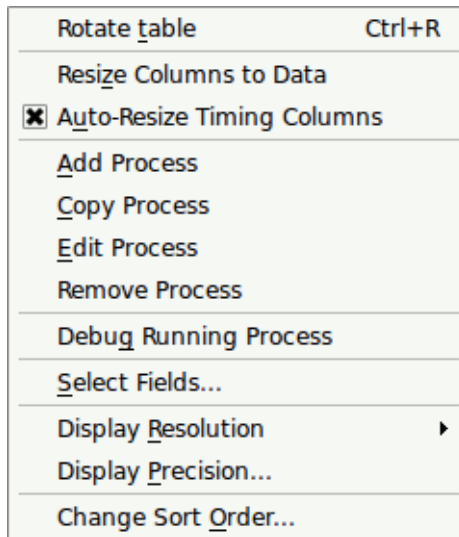


Figure 5-5. The Metrics Panel Context Menu

Rotate Table

Mnemonic: T

This menu option toggles the orientation of the table. By default, the table uses rows for processes and columns for process attributes. When rotated, the rows are used for process attributes and the columns for processes. The latter orientation may be useful when you only are interested in a few process attributes and you have several processes.

Resize Columns To Data

Mnemonic: Z

This menu option resizes all the columns in the table such that the current contents will fit optimally within their columns without truncation and without excessive margins.

Note that columns related to process times may automatically resize based on the Auto-Resize Timings Columns settings discussed below. Other columns do not resize automatically.

Auto-Resize Timing Columns

Mnemonic: U

When checked, this option continually resizes columns related to time values to ensure their entire values can be seen.

Add Process

Mnemonic: A

This menu option launches the Add Process dialog. See “Add Process and Edit Process Dialog Pages” on page 4-22 for more information.

Copy Process

Mnemonic: C

This menu option makes a copy of the process you right-clicked on and adds it to the scheduler.

Edit Process

Mnemonic: E

This menu option launches the Edit Process dialog for the process associated with the row corresponding to the mouse location when the context menu was launched. See “Add Process and Edit Process Dialog Pages” on page 4-22 for more information.

Remove Process

This menu option removes the process you right-clicked on from the scheduler.

Debug Running Process

Mnemonic: G

This menu option invokes the NightView debugger and attaches to the process that you right-clicked on. Once the debugger finishes the attach step, the process is automatically resumed. You can debug it without stopping it (thereby not affecting its timing), or stop it and debug traditionally.

Note that depending on the length of your scheduler’s cycle, the act of attaching to the process may cause it to overrun during attach processing.

Select Fields...

Mnemonic: S

When you select this option, NightSim displays a dialog which allows you to select the monitoring attributes of interest -- these correspond to columns in the table.

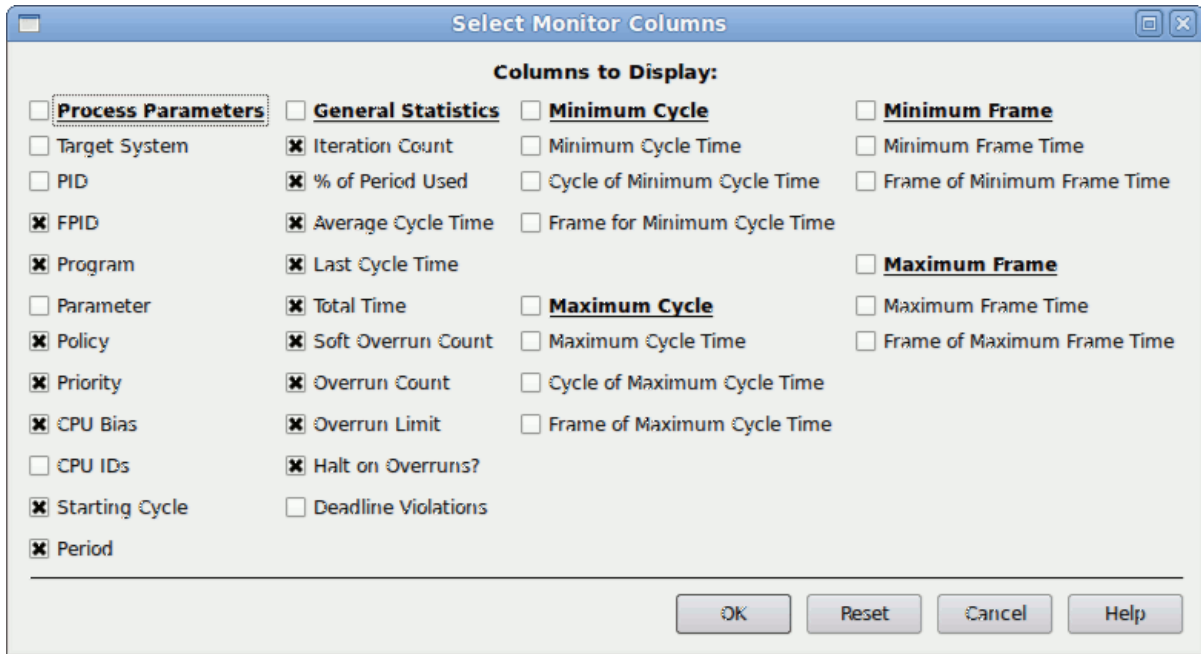


Figure 5-6. Select Monitor Columns Dialog

You use the panels of check buttons to select the types of *performance monitor* values that you wish to save. Click on one or more buttons as desired. For a description of each value, see the section “What Values Are Monitored?” on page 5-3.

Display Resolution

Mnemonic: R

This option allows you to control the precision with which NightSim reports the *performance monitor* timing values. When you select this option, NightSim displays the menu that Figure 5-3 presents.

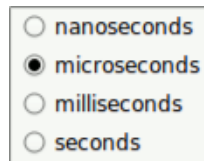


Figure 5-7. Resolution Options

You may specify the precision in nanoseconds, microseconds, milliseconds, or seconds by clicking on a radio button.

Display Precision...

Mnemonic: P

This item allows you to control the number of fractional decimal digits used to display performance values associated with time. It affects minimum, maximum, average, and last cycle times. It does not affect cumulative time values.

Change Sort Order...

Mnemonic: O

This option launches the Sort Key Selection dialog, which allows you to manipulate the sorting order in the Monitor panel.

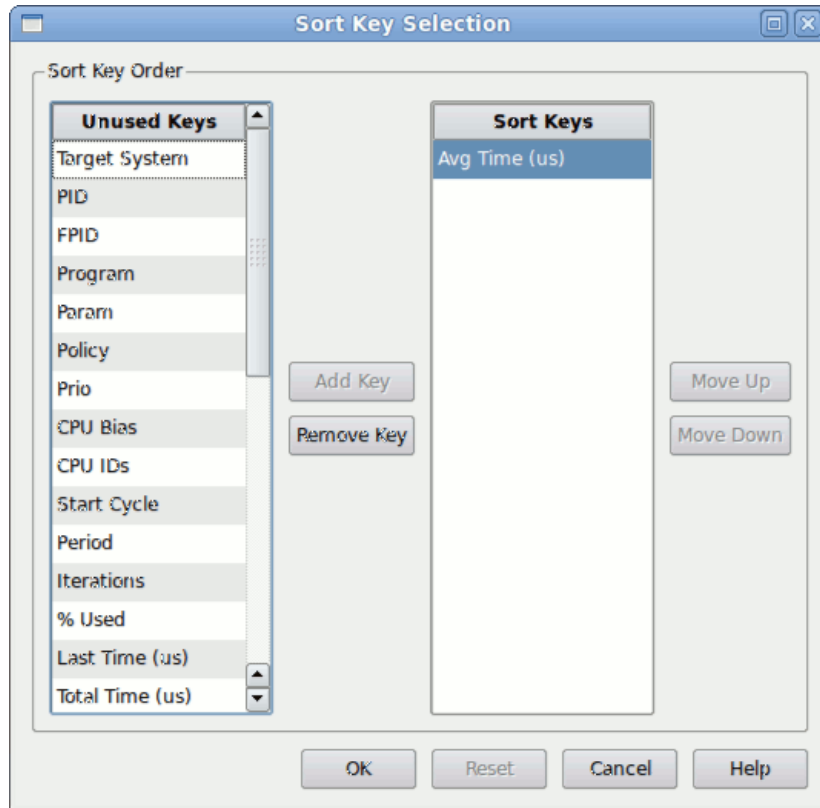


Figure 5-8. Sort Key Selection Dialog

The dialog consists of two columns of sort keys. The column on the left lists keys which are not used during sorting. The column on the right lists the keys consulted for sorting, in order of importance, from top to bottom.

Click the **Left** and **Right** buttons to move selected keys between columns or double click on a key to move it to the other column.

Click the **Up** and **Down** buttons to adjust sort key importance for selected keys. The most important sort key (primary) is used first; other sort keys are consulted to break ties.

The remaining menu options in the context menu are replications of menu options in the Monitor menu discussed in the previous section.

The Percent of Period Used Panel

This panel provides a graph of process CPU time and a table of CPU time details.

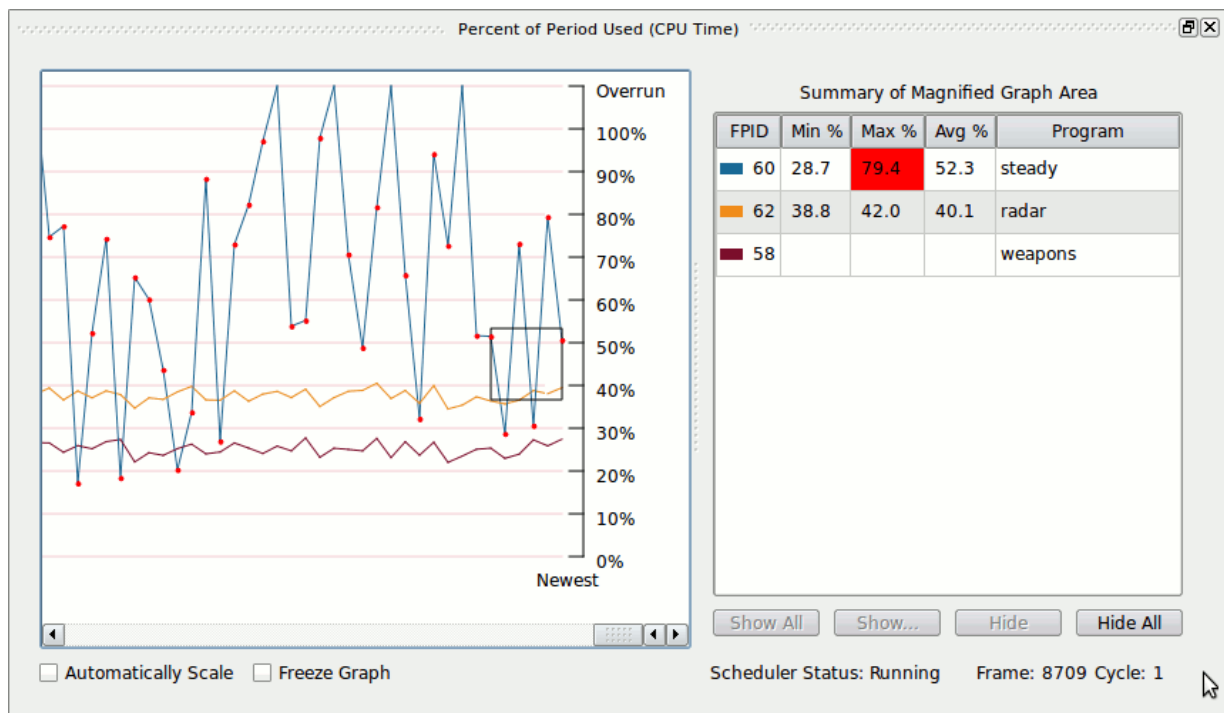


Figure 5-9. Percent of Period Used Panel

Lines on the Graph

By default, the graph shows a single line for each process; although you can omit selected processes from the graph. The colors identify individual processes, as shown in the table to the right. You can change a process' color or omit it from the graph (or subsequently add it back to the graph) using the mouse and buttons from the table on the right.

The points that make up the line are percentage of period used for that process, measured in CPU time. Each percentage calculation is the amount of CPU time used by the process during execution of its cycle, divided by the length of the process' period. For a process that runs every cycle, the period is the length of one cycle as defined by the interrupt device. For a process that runs every other cycle, the period is the length of two such cycles.

The Vertical Axis

The vertical axis represents the percentage calculation.

For CPU times that exceed the process' period, the point is plotted at the top-most vertical coordinate, just above the 100% hash mark, denoted Overrun. In order to make the graph scale well, all such overrun times are plotted at the

same vertical point. However, the table to the right provides additional details (as does the Metrics panel).

Since the graph only plots CPU time, it is possible that the process had over-run its period even though it didn't use all the allotted CPU time. Similarly, the process may have missed its deadline (see "FBS Schedule" on page 4-22 for more information on overruns and deadline violations). If this occurs but the CPU percentage itself does not exceed 100%, a small red dot is shown at the point. Thus all overruns and deadline violations are recorded visually on the graph, either by a red dot or a vertical point exceeding 100%.

The Horizontal Axis

The horizontal axis represents performance monitoring samples over time. The oldest samples are to the left and the newest samples are plotted on the right. Thus the line appears to move to the left over time.

By default, 500 samples are retained for graphing. You can change this value using the graph's context menu.

The Magnifier

The graph also includes a magnification area, which is the black triangle shown in the figure above near the 50% percent hash mark. The magnifier can be dragged to any location on the graph using the mouse. The magnifier controls which data points are shown in the **Summary of Magnified Graph Area** table to the right of the graph. That table shows summary information for each process that contains points that fall within the area of the magnifier.

By default, the magnifier floats above the graph, thus the contents of the table may change at each sample. The table to the right is updated whenever the magnifier is moved or when a new sample is added to the graph. You can pin the magnifier to a specific spot on the graph using the graph's context menu -- when pinned, the magnifier will appear to move to the left when each new sample is added to the graph -- it is pinned to a set of data points.

You can change the size of the magnifier by clicking within its boundaries to show the resize handles. Grab one of the resize handles with the mouse to change the size.

The Summary of Magnified Graph Area Table

The summary table shows includes a row for each scheduled process. Most columns only contain data when the corresponding process has data points in the magnified graph area.

FPID

The process' FBS ID. Double-clicking this cell launches color selection menu which allows you to change the color of a process. User-defined colors are part of the NightSim configuration. When saving a configuration, NightSim attempts to use the same color on subsequent sessions for that process.

Min %, Max %, and Avg %

These columns provide this information only for the processes and the samples that fall within the current geometry of the magnifier. Thus using the magnifier you can examine outlier areas of the graph in more detail.

If a deadline violation or overrun has occurred in the period of time represented by the magnified samples for that process, the background of the Max cell will turn red.

Positioning the mouse cursor over any of these cells and leaving it there for a short period of time will launch a tool tip which describes the summary in more detail:

```
Raw CPU Times (us):
Min: 475.154 @ frame 56657, cycle 1
Max: 509.007 @ frame 57158, cycle 0
Avg: 488.867 over 5002 cycles
Sample Span: ~5.002 seconds
```

Program

This column shows the simple name of the process. Positioning the mouse cursor over this cell and leaving it there for a short period of time will launch a tool tip which describes the process in more detail.

```
Program: /tmp/app1
FPID: 11
PID: 17015
Starting Cycle: 0
Period: 2
Priority: 50
CPU List: 1
Parameter: 0
Deadline: 100000 us
Graph Color: #2927d3
```

Show All, Show..., Hide, and Hide All Buttons

These buttons control whether processes are shown on the graph. Select a process or processes using the mouse in the table and press the **Hide** button to remove them from the graph.

If some processes are hidden, the **Show** button is enabled and pressing it will launch a dialog which allows you to add processes back to the graph.

Automatically Scale

When this option is checked, the entire graph is forced to fit within the available window space.

Freeze Graph

When this option is checked, updates to the graph are delayed until the checkbox is subsequently cleared.

Freezing allows you to interact more easily with the magnifier when it is not floating or when analyzing a section of the graph which may be move when additional samples are added.

Scroll Bars

Scroll bars are shown when there are additional areas of the graph not visible in the available window space. When you interact with the horizontal scroll bar, the graph may no longer show you the newest samples so that you can concentrate without the viewable area moving on you. Dragging the scroll bar all the way to the right again will cause the graph to resume showing you the newest samples.

Mouse Wheel

You can use the mouse wheel to zoom in and out on the graph.

Context Menu

The panel's context menu is shown below:



Figure 5-10. Panel Context Menu

Show/Hide Details

Mnemonic: D

This menu option toggles the visibility of the summary table and the magnifier from the graph panel.

Float/Unfloat Magnifier

This menu option toggles the floating behavior of the magnifier. When floating, the magnifier appears to float over the same location of the graph without moving each time a sample is added. In this mode, the magnifier may continually cover a different set of data points as samples are added.

When not floating, the magnifier is conceptually pinned to a coordinate on the graph, so it appears to move left as samples are added. In this mode, the magnifier always covers the same set of data points until moved or floated again.

Move Magnifier Here

Mnemonic: M

This menu option moves the magnifier to the location of the mouse when the context menu was launched.

You can also move the magnifier by dragging it with the mouse.

Show Magnifier Resize handles

Mnemonic: H

This menu option causes the magnifier's resize handles to appear. Normally, these handles are not shown. You can also simply click within the confines of the magnifier to show the handles.

Once shown, the handles remain visible for 10 seconds before disappearing again.

Change Retained Sample Count

This menu option launches an input dialog which allows you to change the number of samples retained for graphing.

Logging

NightSim can log performance monitoring values to a file. The file format is a comma-separated list of performance monitoring values, with an optional preceding row describing the values.

Logging occurs on the host system (host and target can be the same or different systems). Logging occurs at the interval used to refresh the data in the Monitor panel -- once a second.

Logging is configured through the **Logging** dialog, which can be launched from the **File** menu or using the **Ctrl+L** shortcut key sequence.

Logging Dialog

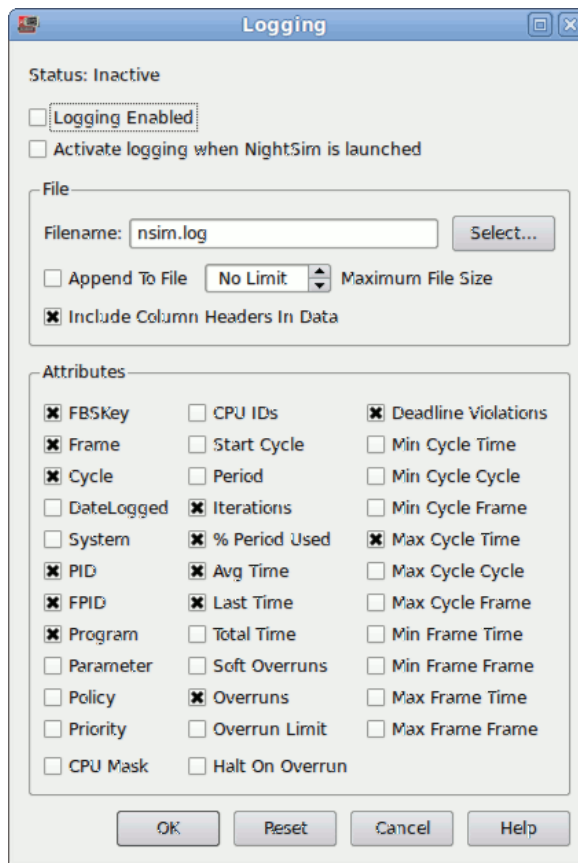


Figure 5-11. Logging Dialog

Logging attributes are part of a NightSim configuration. Changes you make in this dialog are not saved for subsequent use in another NightSim invocation unless you also save the configuration (e.g. File -> Save Config).

Logging has three states, Disabled, Enabled (Ready), and Enabled (Active). Logging isn't actually activated until the scheduler is Set up and Started -- if logging is not disabled and the scheduler is not running, then logging is in the Enabled (Ready) state.

The sections of the dialog are described below.

Enabled

This checkbox controls whether logging is active in the current session. When checked, logging is currently ready or active, or if it was not checked when the dialog was launched, logging will be readied or activated when the OK button is pressed.

Conversely, if you clear the checkbox in the dialog, logging will stop when the OK button is pressed.

Activate logging when NightTune is launched

This checkbox controls whether logging will be automatically enabled when NightSim launches (assuming that you save this configuration for subsequent use).

File Area

Performance data is written to the log file as specified in the **Filename** field.

FileName

The filename can be specified as a relative file name or an absolute filename.

The filename identifies a file which will be written to on the host system when logging is activated.

Append To File

When checked, new logging sessions are appended to the specified file; otherwise, the file is truncated when a logging session begins.

Maximum File Size

This field allows you to set a size limit for the total accumulated output size, in megabytes.

You can use the Up and Down arrows to the right of the field to increment and decrement the value. Specifying a value of zero means there is no limit and the output file may grow large over time.

When a non-zero limit is specified, NightSim will automatically stop logging when the limit is exceeded. A pop-up dialog is shown to indicate this event.

The **Logging Status Bar** uses yellow and red background colors when the current file size approaches 50% and 90% of the size limit, respectively. See “Logging Status Bar” on page 5-24 for more information.

Include Column Headers In Data

When this option is checked, the output file will have a single line that precedes actual data rows which describe each column of data in words. Each description is separated by a comma -- note that the descriptions may contain spaces.

When column headers are included, they are generated each time logging starts; thus if **Append To File** is checked, they will occur at places in the file other than the first line.

Attribute Area

This area contains individual checkboxes for all process attributes that can be logged. All values referring to process time are logged in units of microseconds.

The value of TimeLogged is the value of time in seconds since the Unix epoch (as returned by `time(2)`) -- it is not logged by default.

The default set of enabled attributes include the following:

- FBS Key
- Frame
- Cycle
- PID
- FPID
- Program
- Iterations
- % Period Used
- Avg Time
- Last Time
- Overruns
- Deadline Violations
- Max Cycle Time
- Min Cycle Time

Logging Status Bar

The logging status bar reports on the status of logging and provides you fast access to controls that start, stop, and change logging.



Figure 5-12. Logging Status Bar

The logging status bar is not shown until logging is enabled initially in a NightSim session. Logging is initially enabled via the Logging Dialog (see “Logging Dialog” on page 5-22) or by starting NightSim with a previously-saved configuration file that indicates logging should be activated upon launch.

Status Area

This area describes the status of logging and the size of the logging file. When the logging file begins to approach the file size limit (if any) set in the Logging Dialog, the background changes color. Yellow indicates the size is over 50% of the limit, while red indicates it is over 90% of the limit.

Change...

This button launches the Logging Dialog which allows you to make changes to the logging configuration which aren't otherwise afforded by controls in the logging status bar.

Start/Stop

This button toggles the current status of logging. Pressing the button will cause the logging to transition to the specified state. When the button displays **Start**, pressing it will cause logging to enter the Enabled (Ready) state or Enabled (Active) state, depending on whether the scheduler is active and running. When the button displays **Stop**, pressing it causes logging to enter the disabled state.

NightStar RT 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 (see “License Keys” on page A-1). 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://real-time.ccur.com/support> or by sending an email to support@ccur.com.

Kernel Dependencies

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

The NightStar RT tools can operate in a host-only mode on many different Linux distributions without a RedHawk kernel, cross-targeting to RedHawk systems.

Additionally, the NightStar RT tools can function on such host systems in target mode without the RedHawk kernel, but will lack the numerous advantages afforded by running with it.

Advantages for NightView

The following advantages are afforded NightView when a RedHawk kernel is running:

- Application speed conditions

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

- Signal handling

Allows NightView to pass signals directly to a particular process, avoiding context switching and stopping the process if the signal is handled.

- Branch tracking

Allows NightView to show you a history of branches. This is especially useful for programs that end up in unexpected locations, usually the result of returning from a routine with a corrupted stack frame. The branch history often allows you to locate where the program execution went awry.

Advantages for NightTrace

The following advantages are afforded NightTrace when a 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 real-timekernel 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.

- CUDA Application Tracing

While not specifically a RedHawk kernel feature, RedHawk provides an optimized NVidia driver along with a pre-built NightTrace Illuminator for the CUDA API library. This illuminator automatically instruments user applications that utilize the CUDA API so that you can see all API function entries and returns. This includes the execution of user routines on the GPU itself along with the amount of time spent executing on the GPU.

Advantages for NightProbe

The following advantages are afforded NightProbe when a 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.

The PCI BAR File System is only available with the RedHawk kernel from Concurrent Computer Corporation. On other systems, PCI Device probing will be disabled within NightProbe.

Advantages for NightTune

The following advantages are afforded NightTune when a RedHawk kernel is running:

- Context switch rate

Allows NightTune 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.

- Detailed memory information

Detailed process memory descriptions include the residency and lock state of any page in a process, and their association with physical memory pools for NUMA systems.

- Kernel Activity and Single Process Activity panels

Provides non-intrusive monitoring of kernel or process/thread activity, including percent of time spent in individual routines in the kernel, in shared libraries, and in user processes. Routines are described using their symbolic name.

- Single Process Counter

Provides non-intrusive monitoring of low-level CPU operations, such as cpu cycles, instructions, bus cycles, branches, cache hits and misses, page faults, cpu migrations, and context switches for individual processes/threads.

- CUDA Configuration and Activity

While not specifically a RedHawk kernel feature, RedHawk provides an optimized NVidia driver that allows NightTune to show detailed CUDA configuration information as well as CUDA device activity, including GPU usage, fan speed, GPU memory usage, etc.

Frequency Based Scheduler

The Frequency Based Scheduler is only available on RedHawk systems from Concurrent Computer Corporation. It is required for all NightSim usage.

FBS Process Deadlines are only available for use on RedHawk 5.2.1 and later systems.

On systems without FBS Process Deadline support, the “Apply Deadline” group box will appear shaded and disabled.

NightSim is only included in NightStar distributions intended for use on RedHawk systems.

Glossary

active scheduler window

A Scheduler window that is associated with a scheduler that exists on the system.

CPU bias

The processor or processors on which a program is scheduled to run.

cycle overrun

Another name for a frame overrun.

deadline violation

The condition that occurs when an FBS-scheduled process does not finish its processing before the end of its deadline time.

end-of-cycle scheduling

A form of frequency-based scheduling in which scheduling is triggered when the last process that is scheduled to execute in the current *minor cycle* of the current *major frame* completes its processing.

FBS

Frequency-based scheduler.

FPID

A unique *frequency-based scheduler* process identifier that is returned when a program is scheduled on a frequency-based scheduler.

frame overrun

The condition that occurs when an *FBS*-scheduled process does not finish its processing before it is scheduled to run again.

frequency-based scheduler

A high resolution task synchronization mechanism that enables processes to run at user-specified frequencies.

host system

The system where /usr/bin/nsim runs -- not necessarily the target system where scheduled processes run.

idle time

Time during which the CPU is not busy.

inactive scheduler window

A Scheduler window that is not associated with an existing scheduler.

iteration

One instance of a process's being wakened by a *frequency-based scheduler*.

last time

A value returned by the *performance monitor* indicating the amount of time that an *FBS*-scheduled process has spent running from the last time that it has been wakened by the scheduler until it has called **fbswait**.

major frame

One pass through all of the *minor cycles* with which a *frequency-based scheduler* is configured. A major frame has associated with it a duration, which is obtained by multiplying the duration of a minor cycle by the number of minor cycles per major frame.

maximum cycle cycle

A value returned by the *performance monitor* indicating the number of the *minor cycle* in which the *maximum cycle time* has occurred.

maximum cycle frame

A value returned by the *performance monitor* indicating the number of the *major frame* in which the *maximum cycle time* has occurred.

maximum cycle time

A value returned by the *performance monitor* indicating the greatest amount of time that an *FBS*-scheduled process has spent running in a cycle.

maximum frame frame

A value returned by the *performance monitor* indicating the number of the *major frame* in which the *maximum frame time* has occurred.

maximum frame time

A value returned by the *performance monitor* indicating the greatest amount of time that an *FBS*-scheduled process has spent running during a *major frame*.

minimum cycle cycle

A value returned by the *performance monitor* indicating the number of the *minor cycle* in which the *minimum cycle time* has occurred.

minimum cycle frame

A value returned by the *performance monitor* indicating the number of the *major frame* in which the *minimum cycle time* has occurred.

minimum cycle time

A value returned by the *performance monitor* indicating the least amount of time that an *FBS*-scheduled process has spent running in a cycle.

minimum frame frame

A value returned by the *performance monitor* indicating the number of the *major frame* in which the *minimum frame time* has occurred.

minimum frame time

A value returned by the *performance monitor* indicating the least amount of time that an *FBS*-scheduled process has spent running during a *major frame*.

minor cycle

The smallest unit of frequency maintained by a *frequency-based scheduler*. A minor cycle has associated with it a duration, which is the time that elapses between interrupts generated by the interrupt source that is attached to the scheduler. If the interrupt source is a real-time clock, the minor cycle duration is defined by specifying the number of clock counts per minor cycle and the number of microseconds per clock count.

NightSim host

The system from which NightSim was invoked.

NightSim server

The NightSim server is a program that executes on a *target system* and performs system-level actions on behalf of NightSim. Whenever NightSim needs to perform an operation on a target system, such as creating an *FBS*, scheduling a program on an *FBS*, or starting or stopping an *FBS*, NightSim makes a request of the NightSim server on that target system. NightSim automatically takes care of starting up a NightSim server on a target system when needed.

number of overruns

A value returned by the *performance monitor* indicating the number of times that an *FBS*-scheduled process has caused a soft or hard *frame overrun*.

performance monitor

A mechanism that makes it possible to monitor use of the CPU by processes that are scheduled on a *frequency-based scheduler*.

period

A *frequency-based scheduler* scheduling parameter that specifies the frequency with which a specified program is to be wakened in each *major frame*. A period of one indicates that the program is to be wakened every *minor cycle*; a period of two indicates that it is to be wakened once every two minor cycles; and so on.

privilege

A mechanism through which processes are allowed to perform sensitive operations or override system restrictions.

process dispatch latency

The time that elapses from the occurrence of an external event, which is signified by an interrupt, until the process that is waiting for that external event executes its first instruction in user mode.

process name

The name of the program.

RCIM

Real-Time Clocks and Interrupt Module - an optional hardware module which provides both real-time clocks and edge-triggered interrupt pins for the *host* to which it is attached, allowing those devices to be used as timing and interrupt sources for that host. RCIMs may be connected, and therefore synchronized, via RCIM cables. RCIM real-time clocks and edge-triggered interrupts may be used locally or as distributed timing devices.

scheduler key

A user-supplied numeric identifier for a *frequency-based scheduler*.

server

See *NightSim server*.

soft overrun

Soft overruns are *frame overruns* which give the scheduled process a chance to recover and return to synchronization. When scheduling a process, you can specify a consecutive soft overrun limit count that this process will tolerate and have processed as soft overruns by the kernel. Soft overruns are catastrophic failures only if the process reaches its limit on the number of soft overruns tolerated. This soft overrun limit defaults to 0 for each process.

starting base cycle

A *frequency-based scheduler* scheduling parameter that specifies the first *minor cycle* in which an *FBS*-scheduled process is to be wakened in each *major frame*.

target system

Any system on which processes will be scheduled.

timing mode

The mode under which the *performance monitor* runs. It specifies whether time spent servicing interrupts is to be included in or excluded from performance monitor timing values.

total iterations

A value returned by the *performance monitor* indicating the number of times that an *FBS*-scheduled process has been wakened by the scheduler.

total time

A value returned by the *performance monitor* indicating the total amount of time that an *FBS*-scheduled process has spent running in all cycles.

unscheduled process

A process that is not wakened by the *frequency-based scheduler* and does not call **fbwait**; it is not scheduled to run at a certain frequency.

Symbols

/idle 4-13, 4-20, 4-28, 5-6

A

Accelerators 3-13
Access permissions 4-13
Active Window 4-6, Glossary-1
Add Process dialog 4-19, 4-22
 FBS Schedule Page 4-22
 I/O and Debug Page 4-28
Adding a process to a scheduler 4-17, 4-21, 4-22
Arguments 4-23
Automatic Configuration
 scheduler 4-12
Average time 5-4

C

chmod 4-13
Clearing performance values 5-10, 5-16
Clock period 4-7, 4-14, 4-15
Configuration files
 open 3-4
 save 3-4
 script 3-5
Configuring a scheduler 4-12
Context-sensitive help 3-11
CPU bias 4-27, 5-3, 5-6, Glossary-1
CPU Ids 5-4
cycle overrun Glossary-1
Cycles per frame 4-13

D

Deadline violation 4-5, 4-24, 5-5, Glossary-1
Defining scheduler frequency 4-2-4-3

Deleting a process from a scheduler 4-18
Device Pathname 4-14
DISPLAY variable 3-2

E

Edit Process dialog 4-21, 4-22
 FBS Schedule Page 4-22
 I/O and Debug Page 4-28
Edit processes on a scheduler 4-21, 4-22
End of Processing Cycle 4-2, 4-14
end-of-cycle scheduling 4-2, 4-14, Glossary-1
Environment variable
 DISPLAY 3-2
 NSLM_SERVER A-2
Exit 3-5, 3-23
Exit Immediately 3-6, 3-23
Exiting NightSim 3-5, 3-6, 3-23

F

FBS Schedule Page 4-22
FBS. See Frequency-Based Scheduler
fbswait 5-3, 5-6
File Menu 3-3
File Toolbar 3-12
Fixed licenses A-1
Floating licenses A-1
FPID 5-3, Glossary-1
Frame overruns
 description 4-4, Glossary-1, Glossary-3
 halt 4-24, 5-4, 5-7
 hard 5-4
 limit 4-24, 5-4
 soft 5-4, Glossary-4
Frequency-Based Scheduler
 description 4-1
 overview 1-1
 process identifier 5-3, Glossary-1

G

Graphical user interface (GUI) 1-2

H

Halt on Overrun 4-24, 5-4, 5-7

Hard overruns 5-4

Help Menu 3-10

Host

- NightSim Glossary-3
- scheduler 4-6

I

I/O and Debug Page 4-28

Idle time

- Add /idle to scheduler 4-17, 4-20

- Change CPU bias 4-28

- description Glossary-2

- measuring 4-20

- monitoring 5-6

Inactive Window 3-2, Glossary-2

Interrupt Selection 4-13

Interrupt Source 4-14

Interrupt time

- excluding 5-2, 5-11

- including 5-1, 5-9

Invoking NightSim 3-1

Iteration 5-4, Glossary-2, Glossary-5

K

Key 4-6, 4-12, Glossary-4

Keyboard accelerators 3-13

L

Last time 5-4, Glossary-2

License A-1

- fixed A-1

- installation A-1

- keys A-1

- modes A-1

- ns1m_admin A-1, A-3

- report A-3

- requests A-2

- server A-2

- support A-4

M

Major frame 4-2-4-4, Glossary-2

Max tasks in scheduler 4-13

Max tasks per cycle 4-13

Maximum cycle cycle 5-5, Glossary-2

Maximum cycle frame 5-5, Glossary-2

Maximum cycle time 5-5, Glossary-2

Maximum frame frame 5-5, Glossary-2

Maximum frame time 5-5, Glossary-2

Menu Bar 3-3

- File 3-3

- Help 3-10

- Monitor 3-8

- Scheduler 3-8, 4-5

- Tools 3-8

- View 3-6, 3-7

Minimum cycle cycle 5-5, Glossary-3

Minimum cycle frame 5-5, Glossary-3

Minimum cycle time 5-5, Glossary-3

Minimum frame frame 5-5, Glossary-3

Minimum frame time 5-5, Glossary-3

Minor cycle 4-2-4-4, Glossary-3

Monitor Area 5-1

- example 5-8

Monitor Menu 3-8

Monitoring idle time 5-6

N

New Config 3-4, 3-12

NightSim

- host Glossary-3

- invoking 3-1

- overview 1-1

NightSim server

- description Glossary-3

NightView

- halt on overrun 4-24

ns1m_admin A-1, A-3

NSLM_SERVER A-2

O

On Context menu option 3-11
 Open Config 3-4, 3-12
 Overruns, see Frame overruns
 Overview
 FBS 1-1
 Frequency-Based Scheduler 1-1
 Graphical user interface 1-2
 NightSim 1-1
 Performance Monitor 1-2
 RCIM support 1-4

P

Parameters 4-8, 4-23, 5-4
 Pathname expansion 4-30
 Pause Scheduler 3-13, 4-8, 4-10
 Percent of period used 5-4
 Performance Monitor
 clearing values 5-10
 description 5-1, Glossary-4
 disabled 5-9
 enabled 5-9
 Idle time 5-6
 monitored values 5-3
 overview 1-2
 resolution 5-10
 unscheduled processes 5-6
 Period 4-2-4-5, 4-23, 5-3, 5-4, Glossary-4
 Permissions 4-13
 Policy 4-27, 5-3
 Priority 4-27, 5-3
 Process Dispatch Latency Glossary-4
 Process identifier
 frequency-based scheduler Glossary-1
 process 5-3
 Process Menu 4-17
 Process Scheduling Illustration 4-3
 Processes
 add to scheduler 4-17, 4-22
 PID 5-3
 scheduling 4-2-4-5
 sleep state 4-3
 unscheduled 4-23, 5-6, Glossary-5
 view scheduling data 4-15
 Processes Area 4-15
 Add Process dialog 4-19, 4-22
 Edit Process dialog 4-21, 4-22
 Program Name 5-3
 Programming requirements 4-5

R

Real-Time Clock and Interrupt Module (RCIM) 1-4,
 2-1, 4-14, Glossary-4
 Refresh Rate 4-11
 Remove Scheduler 3-13, 4-8, 4-10
 Removing a process from a scheduler 4-18
 Removing a scheduler 3-13, 4-8, 4-10
 Rescheduling processes 4-15
 Resolution 5-10
 Resume Scheduler 3-13, 4-8, 4-10
 rtcp 3-5

S

Save Config 3-4, 3-12
 Save Config As 3-4
 Save Script As 3-5
 Scheduler
 add process 4-17
 configuration files 3-4
 configuration script 3-5
 configuring 4-12
 cycles per frame 4-13
 edit process 4-17
 FPID 5-3
 frequency 4-2-4-3
 key 4-6, 4-12, Glossary-4
 max tasks in scheduler 4-13
 max tasks per cycle 4-13
 operation 4-3-4-5
 pause 3-13, 4-8, 4-10
 permissions 4-13
 remove 3-13, 4-8, 4-10
 remove process 4-18
 resume 3-13, 4-8
 running simulation 4-11
 select active 3-12, 4-6
 set up 3-12, 4-7, 4-10
 start/restart 3-13, 4-7
 start/resume 4-10
 stopped simulation 4-11
 Scheduler Control Area 4-9
 Scheduler Definition Area 4-12
 Scheduler Interrupt Area 4-13
 Scheduler Menu 3-8, 4-5
 Scheduler Toolbar 3-12
 Scheduling policy 4-27
 Scheduling processes 4-2-4-5, 4-15
 Select Active Scheduler 3-12, 4-6
 Setup Scheduler 3-12, 4-7, 4-10

Simulation

- interrupt source 4-14
- pause 3-13
- refresh rate 4-11
- resume 3-13
- start/stop 3-13
- starting 4-8, 4-10
- state indicator 4-11
- stopping 4-8, 4-10
- Soft overrun limit 4-24
- Soft overruns 5-4, Glossary-4
- Start/Restart Scheduler 3-13, 4-7
- Start/Resume Scheduler 4-10
- Starting a simulation 4-8, 4-10
- Starting Cycle 4-2, 4-22, 5-3, Glossary-5
- Starting NightSim
 - options 3-1
 - steps 3-2
- Stopping a simulation 4-8, 4-10
- System updates 3-11

T

- Target System 4-6, 5-3, Glossary-5
- Timing mode 5-1, Glossary-5
- Timing Source 4-13, 4-14
- Toolbar 3-12
 - File 3-12
 - Scheduler 3-12
- Tools Menu 3-8
- Total iterations 5-4
- Total time 5-4, Glossary-5

U

- Unscheduled processes 4-23, 5-6, Glossary-5
- User-supplied timing device 4-14

V

- View Menu 3-6, 3-7
- Viewing FBS-scheduled processes 4-15

X

- xterm 4-29