NightView

Version 5.3 Release Notes (Linux)

April 2001

0898395-5.3

CONCURRENT
COMPUTER
CORPORATION

Copyright

Copyright 2001 by Concurrent Computer Corporation. All rights reserved. This publication or any part thereof is intended for use with Concurrent Computer Corporation products by Concurrent Computer Corporation personnel, customers, and end–users. It may not be reproduced in any form without the written permission of the publisher.

Disclaimer

The information contained in this document is subject to change without notice. Concurrent Computer Corporation has taken efforts to remove errors from this document, however, Concurrent Computer Corporation's only liability regarding errors that may still exist is to correct said errors upon their being made known to Concurrent Computer Corporation.

License

Duplication of this manual without the written consent of Concurrent Computer Corporation is prohibited. Any copy of this manual reproduced with permission must include the Concurrent Computer Corporation copyright notice.

Trademark Acknowledgments

NightView, PowerWorks, PowerMAXION, PowerMAX OS, TurboHawk, and Power Hawk are trademarks of Concurrent Computer Corporation.

Night Hawk is a registered trademark of Concurrent Computer Corporation.

Motorola is a registered trademark of Motorola, Inc.

PowerStack is a trademark of Motorola, Inc.

Linux is a registered trademark of Linus Torvalds.

Red Hat is a registered trademark of Red Hat, Inc.

Intel is a registered trademark of Intel Corporation.

X Window System is a trademark of The Open Group.

Contents

1.0 Introduction 1

2.0 Documentation 2

3.0 Prerequisites 3
- 3.1 Host System 3
  - 3.1.1 Software 3
  - 3.1.2 Hardware 4
- 3.2 Target System 4
  - 3.2.1 Software 4
  - 3.2.2 Hardware 5

4.0 System Installation 6
- 4.1 Separate Host Installation 6
- 4.2 Target Installation 8

5.0 Overview of NightView 5.3 10
- 5.1 Enhancements 10
  - 5.1.1 Watchpoints 10
  - 5.1.2 Long Long 10
  - 5.1.3 Default Translations 10
  - 5.1.4 Dynamically-linked Programs 10
- 5.2 Changes in This Release 11
  - 5.2.1 Stack Pointer and Program Counter 11
  - 5.2.2 Graphical User Interface 11
  - 5.2.3 Internal Communications 11
  - 5.2.4 Login Command 11
  - 5.2.5 Vectors 11
  - 5.2.6 Selecting Context 11
  - 5.2.7 Dialogue Output 11
  - 5.2.8 Searching 12
  - 5.2.9 Online Help 12
  - 5.2.10 Finish Command 12
  - 5.2.11 Language Support 12

6.0 Cautions 13
- 6.1 SFNOLIM Kernel Tunable 13
- 6.2 Breakpoint Command Streams 13
- 6.3 Fortran 77 Arrays 13
- 6.4 Scroll Bars 13
- 6.5 Accessing Variables While in Prologue of Subprogram 13
- 6.6 Shared Objects 14

7.0 Direct Software Support 15

1.0. Introduction

NightView™ is part of the PowerWorks™ Linux Development Environment (PLDE) and is a general-purpose, graphical source-level debugging and monitoring tool specifically designed for real-time applications written in Ada, C, C++, and Fortran. NightView can monitor, debug, and patch multiple real-time processes running on multiple processors on the local system or on different targets with minimal intrusion. In addition to standard debugging capabilities, NightView supports application-speed eventpoint conditions, hot patches, synchronized data monitoring, exception handling and loadable modules.

NightView 5.3 has these new features:

- Watchpoints: the process stops when it reads or writes a specified memory location.
- Long long support for C and C++.
- Default object filename translations for remote debugging.

2.0. Documentation

Table 2-1 lists the NightView 5.3 documentation available from Concurrent.

Table 2-1. NightView Version 5.3 Documentation

Manual Name	Pub. Number
NightView User's Guide	0890395-220
NightView Version 5.3 Release Notes (Linux)	0898395-5.3
NightView Pocket Reference	0890475-040

Copies of the Concurrent documentation can be ordered by contacting the Concurrent Software Support Center. The toll-free number for calls within the continental United States is 1-800-245-6453. For calls outside the continental United States, the number is 1-954-283-1822 or 1-305-931-2408.

Additionally, the *NightView User's Guide* and *NightView Version 5.3 Release Notes (Linux)* are available:

- online using the PowerWorks Linux Development Environment utility, **`nhelp`**
- in PDF format in the **`documentation`** directory of the PLDE Installation CD
- on the Concurrent Computer Corporation web site at www.ccur.com

3.0. Prerequisites

Prerequisites for NightView Version 5.3 for both the host system and target system are as follows:

3.1. Host System

3.1.1. Software

- Red Hat® Linux*
- Required capabilities

NOTE

The following capabilities are normally installed as part of the standard installation of Red Hat Linux and the PowerWorks Linux Development Environment. During installation of the PLDE, the user will be notified if required capabilities do not exist on the Linux system.

- PowerWorks Linux Development Environment

Capabilities	RPMs providing these capabilities
`plde-HyperHelp` `plde-HyperHelp-scripts`	`plde-x11progs-6.4.2-000` `plde-HyperHelp-scripts-6.4.2-000`

* This product has been extensively tested on Red Hat Linux 6.1 and 6.2. However, this product has not been tested with versions of Linux supplied by other vendors.

- Red Hat Linux

Capabilities	RPMs providing these capabilities
ld-linux.so.2 **libICE.so.6** **libSM.so.6** **libX11.so.6** **libXext.so.6** **libXmu.so.6** **libXp.so.6** **libXt.so.6** **libc.so.6** **libc.so.6(GLIBC_2.0)** **libc.so.6(GLIBC_2.1)** **libm.so.6** **libm.so.6(GLIBC_2.0)** **libncurses.so.4** **libpthread.so.0** **libpthread.so.0(GLIBC_2.0)** **libpthread.so.0(GLIBC_2.1)** **libstdc++-libc6.1-1.so.2**	*Red Hat 6.1*: **XFree86-libs-3.3.5-3** **bash-1.14.7-16** **egcs-c++-1.1.2-24** **glibc-2.1.2-11** **libstdc++-2.9.0-24** **ncurses-4.2-25** *Red Hat 6.2*: **XFree86-libs-3.3.6-20** **bash-1.14.7-22** **egcs-c++-1.1.2-30** **glibc-2.1.3-15** **libstdc++-2.9.0-30** **ncurses-5.0-11**

3.1.2. Hardware

- an Intel®-based PC - 300Mhz or higher (recommended minimum configuration)
- 64MB physical memory (recommended minimum configuration)

3.2. Target System

3.2.1. Software

- PowerMAX OS 4.3 or later

  To use watchpoints, you must have PowerMAX OS 4.3 patch-level 6 (base-006) or PowerMAX OS 5.1. NightView disables the use of watchpoints if it doesn't detect the corresponding support in the kernel.

- **Nviewp** 5.3 (see “Target Installation” on page 8 for more information)

3.2.2. Hardware

- Computer Systems:
  - Power Hawk™ 620 and 640
  - Power Hawk 710, 720 and 740
  - PowerStack™ II and III
  - Night Hawk® Series 6000
  - TurboHawk™
  - PowerMAXION™
- Board-Level Products:
  - Motorola® MVME2604
  - Motorola MVME4604

4.0. System Installation

Installation of the host portion of NightView is normally done as part of the general installation of the PowerWorks Linux Development Environment software suite. A single command installs (or uninstalls) all software components of the PLDE, as described in the *PowerWorks Linux Development Environment Release Notes* (0898000).

The following section describes how to install (or uninstall) NightView separately from the PLDE suite for those cases when this is necessary.

In addition, NightView requires that certain packages be installed on the PowerMAX OS target systems. See “Target Installation” on page 8 for more information.

4.1. Separate Host Installation

At times, it may be necessary to install (or uninstall) NightView independent of the installation of the PowerWorks Linux Development Environment software suite. This may be done using the standard Linux product installation mechanism, **`rpm`** (see **`rpm(8)`**).

The RPM name associated with NightView 5.3 is:

```
plde-NightView
```

and the file associated with this RPM is:

```
plde-NightView-5.3-000.i386.rpm
```

which can be found in the **`linux-i386`** directory on the PowerWorks Linux Development Environment Installation CD.

NOTE

The user must be root in order to use the **`rpm`** product installation mechanism on the Linux system.

To install the NightView RPM, issue the following commands on your Linux system:

1. Insert the PowerWorks Linux Development Environment Installation CD in the CD-ROM drive
2. Mount the CD-ROM drive (assuming the standard mount entry for the CD-ROM device exists in **`/etc/fstab`**)

   ```
   mount /mnt/cdrom
   ```

3. Change the current working directory to the directory containing the NightView RPM

   ```
   cd /mnt/cdrom/linux-i386
   ```

4. Install the RPM

   rpm -i plde-NightView-5.3-000.i386.rpm

   By default, the product is installed in **/usr/opt**. To install in a different directory, add

   --relocate /usr/opt=*directory*

   to the **rpm** command where *directory* is the desired directory.

5. Change the current working directory outside the **/mnt/cdrom** hierarchy

   cd /

6. Unmount the CD-ROM drive (otherwise, you will be unable to remove the PowerWorks Linux Development Environment Installation CD from the CD-ROM drive)

   umount /mnt/cdrom

To uninstall the NightView RPM, use the following command:

rpm -e plde-NightView

4.2. Target Installation

NightView requires that the following software package is installed on the PowerMAX OS target systems:

NightStar Tool	Package	Version
NightView	**Nviewp**	5.3

This package is normally installed during the installation of the corresponding NightStar tool on the PowerMAX OS system. However, the package is also included on the PowerWorks Linux Development Environment Installation CD and may be installed by following the procedures below if the above package has not been installed on the target system, or if the version installed on the target system is not the same as that specified above.

NOTE

The version of the required software package installed on the PowerMAX OS target systems must be the same as that listed above. If this is not the case, follow the procedures below.

If your PowerMAX OS system has a CD-ROM device:

```
mkdir /mnt/tmp
mount -F cdfs /dev/dev_name /mnt/tmp
cd /mnt/tmp/powermax-ppc604
```

If your PowerMAX OS system does not have a CD-ROM device:

On the Linux system, insert the PowerWorks Linux Development Environment Installation CD in the CD-ROM drive, mount the CD-ROM drive, and use NFS to export it to the PowerMAX OS system:

```
mount /mnt/cdrom
exportfs powermax_os_system:/mnt/cdrom
```

On the PowerMAX OS system, mount the CD-ROM as an NFS filesystem:

```
mkdir /tmp/cdrom
mount -F nfs linux_system:/mnt/cdrom /tmp/cdrom
cd /tmp/cdrom/powermax-ppc604
```

On the PowerMAX OS system, install the target portions of the NightStar tools via the following commands:

```
cp ./nightstar.Z /tmp/nightstar.Z
zcat /tmp/nightstar.Z | pkgadd -d -
```

If your PowerMAX OS system has a CD-ROM device:

```
cd /
umount /mnt/tmp
```

If your PowerMAX OS system does not have a CD-ROM device:

On the PowerMAX OS system, unmount the NFS filesystem:

```
cd /
umount /tmp/cdrom
```

On the Linux system, unexport the NFS filesystem, and then unmount the CD-ROM:

```
cd /
exportfs -u powermax_os_system:/mnt/cdrom
umount /mnt/cdrom
```

5.0. Overview of NightView 5.3

5.1. Enhancements

5.1.1. Watchpoints

A watchpoint stops your program when a particular area of memory is read or written. See the section titled “Watchpoints” in Chapter 3 of the *NightView User’s Guide* (0890395-220).

NightView allows you to specify variables or specific memory locations to watch for reads or writes, using hardware features to minimize intrusion on the process. The process stops when the specified location is referenced or is changed, depending on the watchpoint options. Like other eventpoints, watchpoints can be enabled or disabled and can have conditions and ignore counts.

5.1.2. Long Long

NightView now supports the 64-bit integer types available in C and C++ as `long long` and `unsigned long long`.

5.1.3. Default Translations

NightView 5.3 has default object filename translations. This is interesting only for remote debugging. See the section titled “Remote Debugging” in Chapter 3 of the *NightView User’s Guide* (0890395-220).

Filename translations are required when an executable program or shared library isn't accessible via the same filename from the local and remote systems. An example where filename translation is useful is when you are debugging PowerMAX OS programs that use shared libraries from a Linux host system. For example, the target PowerMAX OS executable might require **`/usr/lib/libc.so.1`** which is accessible from the host Linux system as **`/pmax/os/4.3/moto/usr/lib/libc.so.1`**.

The default translations are formed by examining the system mount table and by detecting which cross development environments exist on the local system. In many cases, the default translations are sufficient, so the user does not need to use the **`translate-object-filename`** command. See the section titled "translate-object-filename" in Chapter 7 of the *NightView User’s Guide* (0890395-220).

Earlier releases of NightView used the first translation that matched the object filename. NightView 5.3 goes through the matching translations until it finds a file containing a text segment with the same contents as the text segment of the program executing. If no file is found with the same text segment contents, then NightView gives a warning and uses the first translation that matched the object filename.

5.1.4. Dynamically-linked Programs

If your program is dynamically-linked, NightView runs the process just far enough to complete the dynamic linking. The final step of dynamic linking is to call any initialization routines in each dynamic library. Earlier versions of NightView stopped the process after these initialization routines were complete. NightView 5.3 stops the process just before these routines are called. This allows you to debug the initialization routines. See the section titled “Debugging with Shared Libraries” in Chapter 3 of the *NightView User’s Guide* (0890395-220).

5.2. Changes in This Release

5.2.1. Stack Pointer and Program Counter

NightView now does rudimentary checks of the stack pointer and program counter each time the program stops. NightView gives a warning if either one contains an address which is not part of the process or does not have the appropriate permissions.

5.2.2. Graphical User Interface

NightView 5.3 improves the performance of removing monitorpoints in the graphical user interface.

NightView 5.3 fixes a problem that would sometimes cause the X library to get a "Bad drawable" error with monitorpoints.

NightView 5.3 fixes a problem with opening a second debug window on Linux.

NightView 5.3 fixes a problem with trying to remove the commands attached to a breakpoint from the GUI.

5.2.3. Internal Communications

For a local dialogue, NightView uses shared memory to communicate between the two processes that make up NightView. NightView 5.3 makes these communications more reliable on systems running programs at a high priority.

5.2.4. Login Command

If a **`login`** command fails to connect to the remote system, the command terminates a non-interactive command stream. See the section titled “Command Streams” in Chapter 3 and the section titled "login" in Chapter 7 of the *NightView User’s Guide* (0890395-220).

5.2.5. Vectors

When printing a vector with the **`print`** command, you can now use the **`/f`** option to print the vector as 4 floating point numbers. All other formats print as 4 hexadecimal numbers.

The **`vector-set`** command now works correctly on Linux. See the section titled "vector-set" in Chapter 7 of the *NightView User’s Guide* (0890395-220).

5.2.6. Selecting Context

NightView 5.2 could not select C threads when running on PowerMAX OS 4.3. NightView 5.3 can select C threads on either PowerMAX OS 4.3 or PowerMAX OS 5.1. See the section titled "select-context" in Chapter 7 of the *NightView User’s Guide* (0890395-220).

The **`select-context`** command now works correctly for Ada tasks on Linux.

5.2.7. Dialogue Output

NightView 5.3 fixes a problem where nulls in the dialogue output would truncate the rest of the line.

5.2.8. Searching

NightView 5.3 changes how **`forward-search`** and **`reverse-search`** decide where to begin searching. See the sections titled “forward-search” and "reverse-search" in Chapter 7 of the *NightView User’s Guide* (0890395-220).

5.2.9. Online Help

NightView 5.3 improves the wording of some of the help pages for errors and adds more cross references.

The command **`help info`** takes you to a page with links to all the **`info`** commands.

In earlier releases, you could get help on an individual command by entering the command **`help`** *command*. Now, if the command has a special abbreviation, you can also get help on the command by its abbreviation: **`help`** *abbreviation*. For example, the **`backtrace`** command has a special abbreviation **`bt`**, so you can get help on the **`backtrace`** command with **`help bt`**. See the sections titled “Command Syntax” and "help" in Chapter 7 of the *NightView User’s Guide* (0890395-220).

5.2.10. Finish Command

The **`finish`** command now works correctly for routines called by a patchpoint. See the section titled "finish" in Chapter 7 of the *NightView User’s Guide* (0890395-220).

5.2.11. Language Support

In earlier releases, NightView could sometimes fail to recognize that a function had only inline instances and would let you try to call the function. Calling functions with only inline instances is now disallowed.

In addition, NightView 5.3 fixes various problems in the following areas:

- comparing Ada bit-packed arrays
- subscripting arrays of records in Ada programs
- referring to class members in dynamic_cast<>() expressions
- casting to an enum type
- referring to types nested in classes
- the scoping of Ada renames

NightView 5.3 also fixes a problem that would sometimes cause enums to show duplicate values.

6.0. Cautions

6.1. SFNOLIM Kernel Tunable

The `SFNOLIM` kernel tunable should not be set above 1024; doing so may cause NightView to fail during initialization or when communicating with other tools.

6.2. Breakpoint Command Streams

The current implementation of breakpoint command streams does not include protection against concurrent access of data structures internal to the debugger. Avoid referencing other processes within breakpoint command streams. In practice, this is rarely a problem.

6.3. Fortran 77 Arrays

There is currently a problem with Fortran 77 arrays passed as formal arguments. The compiler generates a temporary variable to optimize references to the array. This temporary variable is then used in place of the original array argument. If the original array argument is not used later in the routine, then the lifetime of the argument is very short and its value is discarded. If you try to use the debugger to reference the argument later in the routine, the results are undefined. You can do two things to avoid running into this problem:

- Use the array later in the routine. For example, pass the array as an argument to another routine. This extends the lifetime of the original argument.
- Use the debugger to look at the array at the first executable line of the routine. The value of the original argument probably still exists at this point.

6.4. Scroll Bars

Some users have had a problem using the scroll bars in the graphical user interface. When the up-arrow button on a scroll bar is clicked, the window scrolls the length of the entire window contents, rather than just one line. This problem is caused by an inappropriate X resource, which has apparently been copied from one user to another in much the same way as **`$HOME/.profile`** files are often copied:

```
*XmScrollBar*translations:   <Btn1Down>:   Select()
```

You should remove this resource from your **`$HOME/.Xdefaults`** or whatever other means you use for setting X resources. This resource affects any Motif application, not just NightView.

6.5. Accessing Variables While in Prologue of Subprogram

Because the compilers generate line-number information for code that comprises the prologue of a subprogram, you may inadvertently set an eventpoint or stop your program at such a location. The prologue is the code that the compiler generates to set up the execution environment for a subprogram; until that code completes execution, the environment is incomplete. Attempts to access variables while in the context of the prologue may result in errors from NightView or may result in erroneous values.

If you suspect such a problem, you can determine whether you are in the prologue by doing the following:

- Stop the program at the point where the referencing problem occurs.
- Enter the command **`info frame`**.
- If the output from this command says something like "Could not obtain frame description information", then you are probably stopped in the prologue.

To try and avoid this problem, do not set an eventpoint on the lines containing a subprogram heading. If you want to set the eventpoint at the beginning of the subprogram, use the *unit_name* (for Ada) or *function_name* form of location specifier, rather than a *file_name* and *line_number*. See the section titled "Location Specifiers" in Chapter 7 of the *NightView User's Guide* (0890395-220).

6.6. Shared Objects

For core files, NightView cannot always find correct stack description information (tdesc) for shared objects. This problem occurs when the shared objects are built without position-independent code. This problem can cause NightView to display the wrong results for any routine other than the most recently called routine. For example, a **`backtrace`** command may print the wrong information, or an **`up`** command may fail. This is true for any release of NightView, not just 5.3. This is not a problem for running processes, only for core files.

We recommend that you build shared objects with position-independent code if you plan to debug programs that use them. For C and Fortran, use the **`-Zpic`** option. See **`cc(1)`** or **`f77(1)`**. For Ada, use **`-sm shared`** or **`-sm both`**. For HAPSE Ada, see "Shared Libraries and Shared-Objects" in the *HAPSE Reference Manual* (0890288). For MAXAda, see "Compile Options" and "Shared Objects" in the *MAXAda Reference Manual* (0890516).

7.0. Direct Software Support

Software support is available from a central source. If you need assistance or information about your system, please contact the Concurrent Software Support Center at 1-800-245-6453. Our customers outside the continental United States can contact us directly at 1-954-283-1822 or 1-305-931-2408. The Software Support Center operates Monday through Friday from 8 a.m. to 7 p.m., Eastern Standard time.

Calling the Software Support Center gives you immediate access to a broad range of skilled personnel and guarantees you a prompt response from the person most qualified to assist you. If you have a question requiring on-site assistance or consultation, the Software Support Center staff will arrange for a field analyst to return your call and schedule a visit.