



---

## Solaris™ on Sun™ Hardware Reference Manual Supplement

---

Sun Microsystems, Inc.  
901 San Antonio Road  
Palo Alto, CA 94303-4900  
U.S.A

Part No.:806-0422-10  
May 1999, Revision A

Send comments about this document to:  
[docfeedback@sun.com](mailto:docfeedback@sun.com)

Copyright 1999 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, California 94303-4900 U.S.A.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, AnswerBook, SunVTS, and Solaris are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 1999 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, California 94303 Etats-Unis. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, AnswerBook, SunVTS, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REpondre A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



# Contents

---

## **Preface v**

### **1. System Administration Commands (1M) 1M-1**

afbconfig 1M-1

GFXconfig 1M-9

rscadm 1M-13

sunvts 1M-17

vtsk 1M-18

vtsprobe 1M-19

vtstty 1M-22

vtsui 1M-24

vtsui.ol 1M-25

### **2. Device and Network Interfaces (7) 7-26**

afb 7-26



# Preface

---

The *Solaris on Sun Hardware Reference Manual Supplement* contains manual pages (man pages) for software provided to Sun hardware customers with the Solaris 7 product. These supplement the man pages provided in the general *Solaris 7 Reference Manual*.

Before you can access some of the information published in this book through the man command, you may need to install software from the Sun Microsystems Computer Systems Supplement CD for your Solaris release. In most cases, when you install a software cluster from the Sun Microsystems Computer Systems Supplement CD, man pages about the software in that cluster will be automatically installed. For information about installing the man page software, refer to the *Solaris 7 Sun Hardware Platform Guide*.

---

## How This Book Is Organized

This manual contains man pages in alphabetical order within each man page category. Supplemental man pages are included for the following categories:

- System Administration Commands (1M)
- Device and Network Interfaces (7)

The man pages apply to the following products:

- AFB graphics accelerator: `afb`, `afbconfig`
- PGX32 frame buffer: `GFXconfig`
- Sun Remote System Control (RSC): `rscadm`
- SunVTS diagnostic software: `sunvts`, `vtsk`, `vtsprobe`, `vtstty`, `vtsui`, `vtsui.ol`

---

## Sun Documentation on the Web

The `docs.sun.com` web site enables you to access Sun technical documentation on the World Wide Web. You can browse the `docs.sun.com` archive or search for a specific book title or subject at `http://docs.sun.com`.

---

## Sun Welcomes Your Comments

We are interested in improving our documentation and welcome your comments and suggestions. You can email your comments to us at `docfeedback@sun.com`. Please include the part number of your document in the subject line of your email.

<b>NAME</b>	afbconfig – configure the AFB Graphics Accelerator
<b>SYNOPSIS</b>	<pre> /usr/sbin/afbconfig [ -dev <i>device-filename</i> ]     [ -res <i>video-mode</i> [ <b>now</b>   <b>try</b> ] [ <b>noconfirm</b>   <b>nocheck</b> ] ]     [ -file <i>machine</i>   <i>system</i> ]     [ -deflinear <b>true</b>   <b>false</b> ]     [ -defoverlay <b>true</b>   <b>false</b> ]     [ -linearorder <b>first</b>   <b>last</b> ]     [ -overlayorder <b>first</b>   <b>last</b> ]     [ -expvis <b>enable</b>   <b>disable</b> ]     [ -sov <b>enable</b>   <b>disable</b> ]     [ -maxwids <i>n</i> ]     [ -extovl <b>enable</b>   <b>disable</b> ]     [ -g <i>gamma-correction-value</i> ]     [ -gfile <i>gamma-correction-file</i> ]     [ -propt ] [ -prconf ] [ -defaults ]  /usr/sbin/afbconfig [ -propt ] [ -prconf ] /usr/sbin/afbconfig [ -help ] [ -res ? ] </pre>
<b>AVAILABILITY</b>	SUNWafbcf
<b>DESCRIPTION</b>	<p><b>afbconfig</b> configures the AFB Graphics Accelerator and some of the X11 window system defaults for AFB.</p> <p>The first form of <b>afbconfig</b> shown in the synopsis above stores the specified options in the OWconfig file. These options will be used to initialize the AFB device the next time the window system is run on that device. Updating options in the OWconfig file provides persistence of these options across window system sessions and system reboots.</p> <p>The second and third forms which invoke only the <b>-prconf</b>, <b>-propt</b>, <b>-help</b>, and <b>-res ?</b> options do not update the OWconfig file. Additionally, for the third form all other options are ignored.</p> <p>Options may be specified for only one AFB device at a time. Specifying options for multiple AFB devices requires multiple invocations of <b>afbconfig</b>.</p> <p>Only AFB-specific options can be specified through <b>afbconfig</b>. The normal window system options for specifying default depth, default visual class and so forth are still specified as device modifiers on the openwin command line (see the Xsun(1) manual page in the Openwindows Reference Manual).</p> <p>The user can also specify the OWconfig file that is to be updated. By default, the machine-specific file in the /etc/openwin directory tree is updated. The <b>-file</b> option can be used to specify an alternate file to use. For example, the system-global OWconfig file in the /usr/openwin directory tree can be updated instead.</p>

Both of these standard OWconfig files can only be written by root. Consequently, the **afbconfig** program, which is owned by the root user, always runs with setuid root permission.

## OPTIONS

- dev** *device-filename*  
Specifies the AFB special file. The default is **/dev/fbs/afb0**.
- file** **machine** | **system**  
Specifies which OWconfig file to update. If **machine**, the machine-specific OWconfig file in the **/etc/openwin** directory tree is used. If **system**, the global OWconfig file in the **/usr/openwin** directory tree is used. If the file does not exist, it is created.
- res** *video-mode* [ **now** | **try** [ **noconfirm** | **nocheck** ] ]  
Specifies the video mode used to drive the monitor connected to the specified AFB device.

The format of these built-in video modes is:

### **widthxheightxrate**

where **width** is the screen width in pixels, **height** is the screen height in pixels, and **rate** is the vertical frequency of the screen refresh. The **s** suffix of **960x680x112s** and **960x680x108s** means that these are stereo video modes. The **i** suffix of **640x480x60i** and **768x575x50i** designates interlaced video timing. If absent, non-interlaced timing will be used. As a convenience, **-res** also accepts formats with '@' (at sign) in front of the refresh rate instead of x. For example: **1280x1024@76**. Note, some video-modes, supported by AFB, may not be supported by the monitor. The list of video-modes supported by the AFB device and the monitor can be obtained by running **afbconfig** with the **-res ?** option (the third form shown in the command synopsis above). A list of all possible video-modes supported on AFB is shown below.

```
1024x768x60
1024x768x70
1024x768x75
1024x768x77
1024x800x84
1152x900x66
1152x900x76
1280x800x76
1280x1024x60
1280x1024x67
1280x1024x76
960x680x112s   (Stereo)
960x680x108s   (Stereo)
640x480x60
640x480x60i    (Interlaced)
768x575x50i    (Interlaced)
```



### Symbolic names

For convenience, some of the above video modes have symbolic names defined for them. Instead of the form **width x height x rate**, one of these names may be supplied as the argument to **-res**. The meaning of the symbolic name **none** is that when the window system is run the screen resolution will be the video mode that is currently programmed in the device.

Name	Corresponding Video Mode
svga	1024x768x60
1152	1152x900x76
1280	1280x1024x76
stereo	960x680x112s
ntsc	640x480x60i
pal	768x575x50i
none	(see text above)

The **-res** option also accepts additional, optional arguments immediately following the video mode specification. Any or all of these may be present.

**now** If present, not only will the video mode be updated in the OWconfig file, but the AFB device will be immediately programmed to display this video mode. (This is useful for changing the video mode before starting the window system).

Note – It is inadvisable to use this suboption with **afbconfig** while the configured device is being used (e.g. while running the window system); unpredictable results may occur. To run **afbconfig** with the **now** suboption, first bring the window system down. If the **now** suboption is used within a window system session, the video mode will be changed immediately, but the width and height of the affected screen won't change until the window system is exited and reentered again. In addition, the system may not recognize changes in stereo mode. Consequently, this usage is strongly discouraged.

**noconfirm** Using the **-res** option, the user could potentially put the system into an usable state, a state where there is no video output. This can happen if there is ambiguity in the monitor sense codes for the particular code read. To reduce the chance of this, the default behavior of **afbconfig** is to print a warning message to this effect and to prompt the user to find out if it is okay to continue. The **noconfirm** option instructs **afbconfig** to bypass this confirmation and to program the requested video mode anyway. This option is useful when **afbconfig** is being run from a shell script.

**nocheck** If present, the normal error checking based on the monitor sense code (described above) will be suspended. The video mode specified by the user will be accepted regardless of whether it is

appropriate for the currently attached monitor. (This option is useful if a different monitor is to be connected to the AFB device). *Use of this option implies noconfirm well.*

**try** If present, the specified video mode will be programmed on a trial basis. The user will be asked to confirm the video mode by typing 'y' within 10 seconds. Or the user may terminate the trial before 10 seconds are up by typing any character. Any character other than 'y' or carriage return is considered a no and the previous video mode will be restored and **afbconfig** will not change the video mode in the OWconfig file (other options specified will still take effect). If a carriage return is typed, the user is prompted for a yes or no answer on whether to keep the new video mode. This option implies the now suboption (see the warning note on the now suboption).

**AFB possesses two types of visuals: linear and nonlinear.**

Linear visuals are gamma corrected and nonlinear visuals are not. There are two visuals that have both linear and nonlinear versions: 24-bit TrueColor and 8-bit StaticGray.

If true, the default visual is set to the linear visual that satisfies other specified default visual selection options (specifically, the Xsun(1) defdepth and defclass options described in the OpenWindows Reference Manual).

If false, or if there is no linear visual that satisfies the other default visual selection options, the non-linear visual specified by these other options will be chosen to be the default.

This option cannot be used when the **-defoverlay** option is present, because AFB doesn't possess a linear overlay visual.

**-defoverlay true | false**

The AFB provides an 8-bit PseudoColor visual whose pixels are disjoint from the rest of the AFB visuals. This is called the overlay visual. Windows created in this visual will not damage windows created in other visuals. The converse, however, is not true. Windows created in other visuals will damage overlay windows. The number of colors available to the windows created using this visual depends on the settings for the extovl option. If the extovl is enabled, extended overlay with 256 opaque color values is available. (refer to the **-extovl** option). If extovl is disabled, extended overlay is not available and this visual has (256 - maxwids) number of opaque color values (refer to the **-maxwids** option).

If the value of this option is true, the overlay visual will be made the default visual.

If false, the nonoverlay visual that satisfies the other default visual selection options, such as defdepth and defclass, will be chosen as the default visual. See the Xsun(1) manual page in the OpenWindows Reference Manual.

Whenever **-defoverlay true** is used, the default depth and class chosen on the openwin command line must be 8-bit PseudoColor. If not, a warning message

will be printed and the **-defoverlay** option will be treated as false.

This option cannot be used when the **-deflinear** option is present, because AFB doesn't possess a linear overlay visual.

**-linearorder first | last**

If true, linear visuals will come before their non-linear counterparts on the X11 screen visual list for the AFB screen. If false, the nonlinear visuals will come before the linear ones.

**-overlayorder first | last**

If true, the depth 8 PseudoColor Overlay visual will come before the non-overlay visual on the X11 screen visual list for the AFB screen. If false, the non-overlay visual will come before the overlay one.

**-expvis enable | disable**

If enabled, OpenGL Visual Expansion will be activated. Multiple instances of selected visual groups (8-bit PseudoColor, 24-bit TrueColor ... etc) can be found in the screen visual list.

**-sov enable | disable**

If enabled, the root window's `SERVER_OVERLAY_VISUALS` property will be advertised. SOV visuals will be exported and their transparent types, values and layers can be retrieved through this property. If disabled, the `SERVER_OVERLAY_VISUALS` property will not be defined. SOV visuals will not be exported.

**-maxwids *n***

This option is available only if `extovl` is disabled. It specifies the maximum number of AFB X channel pixel values that are reserved for use as window IDs (WIDs). The remainder of the pixel values in overlay colormaps are used for normal X11 opaque color pixels.

The reserved WIDs are allocated on a first-come first-serve basis by 3D graphics windows (such as XGL), MBX windows, and windows that have a non-default visual.

The X channel codes 0 to  $(255 - n)$  will be opaque color pixels. The X channel codes  $(255 - n + 1)$  to 255 will be reserved for use as WIDs. Legal values: 1, 2, 4, 8, 16, 32, 64

**-extovl enable | disable**

If enabled, extended overlay is available. The overlay visuals will have 256 opaque colors. The SOV visuals will have 255 opaque colors and 1 transparent color. Also, this option enables hardware supported transparency, thus provides better performance for windows using the SOV visuals.

**-g *gamma-correction value***

This option allows changing the gamma correction value. All linear visuals provide gamma correction. By default the gamma correction value is 2.22. Any value less than zero is illegal.

This option can be used while the window system is running. Changing the gamma correction value will affect all the windows being displayed using the

linear visuals.

**-gfile** *gamma-correction file*

This option loads gamma correction table from the specified file. This file should be formatted to provide the gamma correction values for R, G and B channels on each line. Each of these values should be in hexadecimal format and separated from each other by at least 1 space. Also this file should provide 256 such triplets. An example of this file is as follows.

```
0x00 0x00 0x00
0x01 0x01 0x01
0x02 0x02 0x02
...
...
0xff 0xff 0xff
```

Using this option, the gamma correction table can be loaded while the window system is running. The new gamma correction will affect all the windows being displayed using the linear visuals. Note, when gamma correction is being done using user specified table, the gamma correction value is undefined.

By default, the window system assumes a gamma correction value of 2.22 and loads the gamma table it creates corresponding to this value.

**-defaults**

Resets all option values to their default values.

**-propt** Prints the current values of all AFB options in the OWconfig file specified by the **-file** option for the device specified by the **-dev** option. Prints the values of options as they will be in the OWconfig file after the call to **afbconfig** completes. This is a typical display:

```
--- OpenWindows Configuration for /dev/fbs/afb0 ---
OWconfig: machine
Video Mode: 1280x1024x76
Default Visual: Non-Linear Normal Visual
Visual Ordering: Linear Visuals are last
                  Overlay Visuals are last
OpenGL Visual Expansion: enabled
Server Overlay Visuals: enabled
Extended Overlay: enabled
Underlay WIDs: 64 (not configurable)
Overlay WIDs: 4 (not configurable)
Gamma Correction Value: 2.220000
Gamma Correction Table: Available
```

**-prconf**

Prints the AFB hardware configuration. This is a typical display:

```
--- Hardware Configuration for /dev/fbs/afb0 ---
```

**Type: double-buffered AFB with Z-buffer**  
**Board: rev 0 (Horizontal)**  
**Number of Floats: 6**  
**PROM Information: @(#)afb.fth x.xx xx/xx/xx**  
**AFB ID: 0x101df06d**  
**DAC: Brooktree 9070, version 1 (Pac2)**  
**3DRAM: Mitsubishi 130a, version x**  
**EDID Data: Available - EDID version 1 revision x**  
**Monitor Sense ID: 4 (Sun 37x29cm RGB color monitor)**  
**Monitor possible resolutions: 1024x768x77, 1024x800x84, 1152x900x66,**  
**1152x900x76, 1280x1024x67, 1280x1024x76, 960x680x112s, 960x680x108s**  
**Current resolution setting: 1280x1024x76**

**-help** Prints a list of the **afbconfig** command line options, along with a brief explanation of each.

## DEFAULTS

For a given invocation of **afbconfig** command line if an option does not appear on the command line, the corresponding OWconfig option is not updated; it retains its previous value.

When the window system is run, if an AFB option has never been specified via **afbconfig**, a default value is used. The option defaults are as follows:

Option	Default
-dev	/dev/fbs/afb0
-file	machine
-res	none
-deflinear	false
-defoverlay	false
-linearorder	last
-overlayorder	last
-expvis	enabled
-sov	enabled
-maxwids	32
-extovl	enabled
-g	2.22

The default for the **-res** option of none means that when the window system is run the screen resolution will be the video mode that is currently programmed in the device.

Note – This provides compatibility for users who are used to specifying the device resolution through the PROM. On some devices (e.g. GX) this is the only way of specifying the video mode. This means that the PROM ultimately determines the default AFB video mode.

## EXAMPLES

The following example switches the monitor type to the resolution of 1280 × 1024 at 76 Hz:

```
example% /usr/sbin/afbconfig -res 1280x1024x76
```

**FILES** /dev/fbs/afb0 device special file

**SEE ALSO** mmap(2), fbio(7I), afb(7D)

<b>NAME</b>	GFXconfig – configure the Raptor GFX Graphics Accelerator
<b>SYNOPSIS</b>	<pre> /usr/sbin/GFXconfig [ -dev <i>device-filename</i> ]     [ -res <i>video-mode</i> [ <b>try</b>   <b>noconfirm</b>   <b>nocheck</b> ] ]     [ -file <b>machine</b>   <b>system</b> ]     [ -depth 8 24 ]     [ -defaults ]  /usr/sbin/GFXconfig [ -propt ] [ -prconf ] /usr/sbin/GFXconfig [ -help ] [ -res ? ] /usr/sbin/GFXconfig [ -vpd ] </pre>
<b>DESCRIPTION</b>	<p><b>GFXconfig</b> configures the Raptor GFX Graphics Accelerator and some of the X11 window system defaults for Raptor GFX.</p> <p>The first form of <b>GFXconfig</b> shown in the synopsis above stores the specified options in the OWconfig file. These options will be used to initialize the Raptor GFX device the next time the window system is run on that device. Updating options in the OWconfig file provides persistence of these options across window system sessions and system reboots.</p> <p>The second and third and fourth forms which invoke only the <b>-prconf</b>, <b>-propt</b>, <b>-help</b>, <b>-res?</b> and <b>-vpd</b> options do not update the OWconfig file. Additionally, for the third form all other options are ignored.</p> <p>Options may be specified for only one Raptor GFX device at a time. Specifying options for multiple Raptor GFX devices requires multiple invocations of <b>GFXconfig</b>.</p> <p>Only Raptor GFX-specific options can be specified through <b>GFXconfig</b>. The normal window system options for specifying default depth, default visual class and so forth are still specified as device modifiers on the openwin command line (see Xsun(1)).</p> <p>The user can also specify the OWconfig file that is to be updated. By default, the machine-specific file in the <b>/etc/openwin</b> directory tree is updated. The <b>-file</b> option can be used to specify an alternate file to use. For example, the system-global OWconfig file in the <b>/usr/openwin</b> directory tree can be updated instead.</p> <p>Both of these standard OWconfig files can only be written by root. Consequently, the <b>GFXconfig</b> program, which is owned by the root user, always runs with <b>setuid</b> root permission.</p>
<b>OPTIONS</b>	<p><b>-dev</b> <i>device-filename</i>  Specifies the Raptor GFX special file. The default is <b>/dev/fbs/gfxp0</b>, or <b>/dev/fbs/raptor0</b> if applicable.</p> <p><b>-file</b> <b>machine</b>   <b>system</b>  Specifies which OWconfig file to update. If <b>machine</b>, the machine-specific</p>

OWconfig file in the `/etc/openwin` directory tree is used. If system, the global OWconfig file in the `/usr/openwin` directory tree is used. If the file does not exist, it is created.

**-res** *video-mode* [ **try** | **noconfirm** | **nocheck** ]

Specifies the video mode used to drive the monitor connected to the specified Raptor GFX device.

The format of these built-in video modes is:

**widthxheightxrate**

where width is the screen width in pixels, height is the screen height in pixels, and rate is the vertical frequency of the screen refresh. As a convenience, -res also accepts formats with @ in front of the refresh rate instead of x. For example: 1280x1024@76. The list can be obtained by running GFXconfig with the -res ? option (the third form shown in the command synopsis above). Note that not all resolutions are supported by both the video board and by the monitor. GFXconfig will not permit you to set a resolution the board does not support, and will request confirmation before setting a resolution the monitor does not support.

**Symbolic names**

For convenience, some of the video modes have symbolic names defined for them. Instead of the form widthxheightxrate, one of these names may be supplied as the argument to -res. The meaning of the symbolic name none is that when the window system is run the screen resolution will be the video mode that is currently programmed in the device.

Name	Corresponding Video Mode
<b>svga</b>	<b>1024x768x60</b>
<b>1152</b>	<b>1152x900x76</b>
<b>1280</b>	<b>1280x1024x76</b>
<b>vga</b>	<b>640x480x60</b>
<b>none</b>	(see text above)

The **-res** option also accepts additional, optional arguments immediately following the video mode specification. Any or all of these may be present.

**noconfirm**

Using the **-res** option, the user could potentially put the system into an unusable state, a state where there is no video output. This can happen if there is ambiguity in the monitor sense codes for the particular code read. To reduce the chance of this, the default behavior of GFXconfig is to print a warning message to this effect and to prompt the user to find out if it is okay to continue. The **noconfirm** option instructs **GFXconfig** bypass this confirmation and to program the requested video mode anyway. This option is useful when **GFXconfig** is being run from a shell script.

**nocheck**



If present, the normal error checking based on the monitor sense code (described above) will be suspended. The video mode specified by the user will be accepted regardless of whether it is appropriate for the currently attached monitor. (This option is useful if a different monitor is to be connected to the Raptor GFX device). Use of this option implies noconfirm as well.

**try** If present, the specified video mode will be programmed on a trial basis. The user will be asked to confirm the video mode by typing pattern is displayed. If any character other than prompted for a yes or no answer on whether to keep the new video mode.

**-defaults**

Resets all option values to their default values.

**-propt** Prints the current values of all Raptor GFX options in the OWconfig file specified by the **-file** option for the device specified by the **-dev** option. Prints the values of options as they would be in the OWconfig file after the call to GFXconfig would have completed. This is a typical display:

```
--- OpenWindows Configuration for /dev/fbs/gfxp0 ---
OWconfig: machine
Video Mode: not set
```

**-prconf**

Prints the Raptor GFX hardware configuration. This is a typical display:

```
--- Hardware Configuration for /dev/fbs/gfxp0 ---
DAC: version 0x0
Type:
Board:
PROM: version 0x0
PROM Information:
RAM:
EDID Data:
Monitor Sense ID:
Card possible resolutions: 640x480x60, 800x600x75, 1024x768x60
1024x768x70, 1024x768x75, 1280x1024x75, 1280x1024x76
1280x1024x60, 1152x900x66, 1152x900x76, 1280x1024x67
960x680x112S, 960x680x108S, 640x480x60i, 768x575x50i,
1280x800x76, 1440x900x76, 1600x1000x66, 1600x1000x76,
vga, svga, 1152, 1280, stereo, ntsc, pal
Monitor possible resolutions: 720x400x70, 720x400x88, 640x480x60
640x480x67, 640x480x72, 640x480x75, 800x600x56,
800x600x60, 800x600x72, 800x600x75, 832x624x75,
1024x768x87, 1024x768x60, 1024x768x70, 1024x768x75,
1280x1024x75, 1280x1024x76, 1152x900x66, 1152x900x76,
1280x1024x67, 960x680x112S, vga, svga, 1152, 1280
```

**stereo**

**Current resolution setting: 1280x1024x76**

**Possible depths:**

**Current depth: 8**

- help** Prints a list of the **GFXconfig** command line options, along with a brief explanation of each.
- vpd** Prints a list of the card's Vital Product Data information including serial number and Prom Revision level.

## DEFAULTS

For a given invocation of **GFXconfig**, if an option does not appear on the command line, the corresponding **OWconfig** option is not updated; it retains its previous value, except for **'-depth'** and **'-24only'**.

When the window system is run, if an Raptor **GFX** option has never been specified via **GFXconfig**, a default value is used. The option defaults are as follows:

Option	Default
<b>-dev</b>	<b>/dev/fbs/gfxp0</b>
<b>-file</b>	<b>machine</b>
<b>-res</b>	<b>none</b>

The default for the **-res** option of **none** means that when the window system is run the screen resolution will be the video mode that is currently programmed in the device.

Note - This provides compatibility for users who are used to specifying the device resolution through the PROM. On some devices (e.g. **GX**) this is the only way of specifying the video mode. This means that the PROM ultimately determines the default Raptor **GFX** video mode.

## EXAMPLES

The following example switches the monitor type to the resolution of **1280 x 1024** at **76 Hz**:

```
example% /usr/sbin/GFXconfig -res 1280x1024x76
```

## FILES

<b>/dev/fbs/gfxp0</b>	device special file
<b>/usr/openwin/server/etc/OWconfig</b>	System config file
<b>/etc/openwin/server/etc/OWconfig</b>	Machine config file

<b>NAME</b>	rscadm – administer SUN(tm) Remote System Control (RSC)
<b>SYNOPSIS</b>	<pre> rscadm help rscadm resetrsc [-s] rscadm set <i>variable value</i> rscadm download [boot] <i>file</i> rscadm show [variable] rscadm date [-s]   [[mmdd]HHMM   mmddHHMM[cc]yy][.SS] rscadm send_event [-c] <i>message</i> rscadm modem_setup rscadm useradd <i>username</i> rscadm userdel <i>username</i> rscadm usershow [username] rscadm userpassword <i>username</i> rscadm userperm <i>username</i> [cuar] </pre>
<b>DESCRIPTION</b>	<p>rscadm administers the SUN(tm) Remote System Console (RSC). It allows the host server to interact with the RSC. The following operations are supported:</p> <p><b>rscadm help</b>  Displays a usage screen.</p> <p><b>rscadm resetrsc</b>  Reset the RSC. There are two types of reset allowed, a "hard" reset and a "soft" reset. The hard reset is done by default. The soft reset can be selected by using the -s option.</p> <p><b>rscadm set</b>  Set RSC configuration variables. Examples of RSC configuration variables include RSC IP address and RSC hostname. See the RSC documentation for a complete list of RSC configuration variables.</p> <p><b>rscadm download</b>  Program the RSC's firmware. There are two parts to the firmware, the boot monitor and the main image. By default, rscadm download programs the main firmware image. The boot option selects programming of the boot monitor.</p> <p><b>rscadm show</b>  View the current RSC configuration variable settings. If no variable is specified, rscadm shows all variable settings.</p> <p><b>rscadm date</b>  Show or set RSC's time and date. The -s options can be used to set RSC's time and date to the hosts time and date.</p> <p><b>rscadm send_event</b>  Send a text based event to RSC. RSC may forward the event based on its event configuration.</p>

**rscadm modem\_setup**

Direct connection to the RSC modem. This allows the user to enter AT commands to configure the modem. "~." returns to prompt.

**rscadm useradd**

Add user account to RSC. RSC can support up to four separate users.

**rscadm userdel**

Delete a user account from RSC.

**rscadm usershow**

Show details on the specified user account. If a username is not specified, all user accounts will be shown.

**rscadm userpassword**

Set a password for the user account specified. This password overrides any existing password currently set. There is no verification of the old password before setting the new password. See the RSC documentation on valid password formats.

**rscadm userperm**

Set the authorization profile for the user. See the userperm options section in this man page for more detail.

**OPTIONS**

The following options are supported for rscadm:

**rscadm resetrsc**

**[-s]** Perform a "soft" reset instead of a "hard" reset. A hard reset physically resets the RSC hardware. The RSC software jumps to the boot firmware, simulating a reset, for a soft reset.

**rscadm download**

**[boot]** Program the boot monitor portion of the flash. The main portion of the flash is usually programmed.

**rscadm show**

**[variable]** Show the value of that particular variable.

**rscadm date**

**[-s]** Set the date to the hosts time and date.

**[[mmdd]HHMM | mmddHHMM[cc]yy][.SS]**  
the date.

**mm** - month

**dd** - day

**HH** - hour

**MM** - minute

**cc** - the first two digits of the four digit year

**yy** - last 2 digits of the year number

**SS** - seconds

**rscadm send\_event**

**[-c]** Send a critical event. Without the **-c**, **send\_event** sends a warning. Warnings are only logged in the RSC event log and not forwarded further.

**rscadm usershow**

**[username]**

RSC account name to display info on. If no username is given, all accounts will be displayed.

**rscadm userperm**

**[cuar]** Set permissions for RSC account. If no permissions are specified, all four permissions will be disabled. The options are **t**o; allow user to connect to (c)onsole, allow user to use the (u)ser commands to modify RSC accounts, allow user to (a)dmnister/change the RSC configuration variables, allow the user to (r)eset RSC and to power on/off the host.

**OPERANDS**

The following operands are supported for **rscadm**:

**rscadm set**

*variable* RSC configuration variable to set. See the RSC documentation for a list of configuration variables.

*value* Value to set RSC configuration variable to. See the RSC documentation for a list of valid values.

**rscadm download**

*file* Firmware file to download. The file should contain the RSC boot monitor image or RSC main image.

**rscadm send\_event**

*message* Text message to describe event. Should be enclosed in quotes.

**rscadm useradd**

*username* Username for new RSC account.

**rscadm userdel**

*username* RSC account to be removed.

**rscadm userpassword**

*username* RSC account to have password set.

**rscadm userperm**

*username* RSC account to have permissions changed.

**EXIT STATUS**

**= 0** on success

**!= 0** on failure (with status message)

**EXAMPLES**

```
# rscadm date
# rscadm date -s
# rscadm date 050113101998
```

```
# rscadm set hostname rsc15
# rscadm show
# rscadm show hostname
# rscadm send_event -c "The UPS signaled a loss in power!"
# rscadm send_event "The disk is close to full capacity"
# rscadm useradd rscroot
# rscadm userdel olduser
# rscadm usershow
# rscadm usershow rscroot
# rscadm userperm rscroot cuar
# rscadm userperm newuser c
# rscadm userperm newuser
```

**NOTES** rscadm modem\_setup - "~." will only work after a new line.  
rscadm MUST be run as root.

**BUGS** None known.

<b>NAME</b>	sunvts – Invokes the SunVTS kernel and its user interface
<b>SYNOPSIS</b>	<b>sunvts</b> [ <b>-lepqstv</b> ] [ <b>-o</b> <i>option_file</i> ] [ <b>-f</b> <i>log_dir</i> ] [ <b>-h</b> <i>hostname</i> ]
<b>AVAILABILITY</b>	<b>SUNWvts</b>
<b>DESCRIPTION</b>	The <b>sunvts</b> command is used to invoke the SunVTS user interface and kernel on the same system. It could be used to start the user interface on the local system and connect to the SunVTS kernel on the remote system. By default, it displays CDE Motif graphic interface for CDE environment, OpenLook graphic interface for OpenWindows environment, or TTY interface for non-windowing system.
<b>OPTIONS</b>	<ul style="list-style-type: none"> <li><b>-l</b> Displays SunVTS OpenLook graphic interface.</li> <li><b>-e</b> Disables the security checking feature.</li> <li><b>-f</b> <i>log_dir</i> Specifies an alternative log_file directory. The default log_file directory is <b>/var/opt/SUNWvts/logs</b>.</li> <li><b>-h</b> <i>hostname</i> Starts the SunVTS user interface on the local system, which connects to or invokes the SunVTS kernel on the specified host after security checking succeeds.</li> <li><b>-o</b> <i>option_file</i> Starts the SunVTS kernel with the test options loaded from the specified <i>option_file</i>, which by default is located in <b>/var/opt/SUNWvts/options</b>.</li> <li><b>-p</b> Starts the SunVTS kernel <b>vtsk (1M)</b> such that it does not probe the test system's devices.</li> <li><b>-q</b> Automatically quits both the SunVTS kernel and the user interface when testing stops.</li> <li><b>-s</b> Automatically starts testing from a selected group of tests. The flag must be used with the <b>-o</b> <i>option_file</i> flag.</li> <li><b>-t</b> Starts <b>vts tty (1M)</b>, a TTY based interface, instead of CDE or OpenLook interface.</li> <li><b>-v</b> Displays version information from <b>vtsui(1M)</b> and <b>vtsk(1M)</b>.</li> </ul>
<b>NOTES</b>	If <b>vtsk (1M)</b> is already running on the test system, the <b>sunvts</b> command ignores the <b>-e</b> , <b>-o</b> , <b>-f</b> , <b>-q</b> , <b>-p</b> , and <b>-s</b> options.
<b>SEE ALSO</b>	<b>vtsk(1M)</b> , <b>vts tty(1M)</b> , <b>vtsui(1M)</b> , <b>vtsui.ol(1M)</b> , <b>vtsprobe(1M)</b>

<b>NAME</b>	vtsk – SunVTS diagnostic kernel
<b>SYNOPSIS</b>	<b>vtsk</b> [ <b>-epqsv</b> ] [ <b>-o options_file</b> ] [ <b>-f logfile_directory</b> ]
<b>AVAILABILITY</b>	SUNWvts
<b>DESCRIPTION</b>	<p>The <b>vtsk</b> command starts up the SunVTS diagnostic kernel as a background process. There can only be one copy of <b>vtsk</b> running at a time. Only the superuser can execute this command.</p> <p>Normally, <b>vtsk</b> is automatically started up by the <b>sunvts (1M)</b> command if it is not already running. <b>vtsk</b> will also be invoked by <b>inetd (1M)</b> when there is a connection request from vtsui or vtsui.ol. In that case, the security file, <b>.sunvts_sec</b>, will be checked for the permission before running vtsk on the target host specified by <b>vtsui(1M)</b> or <b>vtsui.ol(1M)</b>.</p>
<b>OPTIONS</b>	<ul style="list-style-type: none"> <li><b>-e</b> Enables the security checking for all connection requests.</li> <li><b>-p</b> Starts SunVTS diagnostic kernel, but does not probe system configuration.</li> <li><b>-q</b> Quits both the SunVTS diagnostic kernel and the attached User Interfaces when the testing is completed.</li> <li><b>-s</b> Runs enabled tests immediately after started.</li> <li><b>-v</b> Display SunVTS diagnostic kernel's version information only.</li> <li><b>-o options_file</b> Starts the SunVTS diagnostic kernel and sets the test options according to the option file named <i>options_file</i>.</li> <li><b>-f logfile_directory</b> Specifies an alternative logfile directory, other than the default.</li> </ul>
<b>EXIT STATUS</b>	<p>The following exit values are returned:</p> <ul style="list-style-type: none"> <li><b>0</b> Successful completion.</li> <li><b>-1</b> An error occurred.</li> </ul>
<b>FILES</b>	<ul style="list-style-type: none"> <li><b>/var/opt/SUNWvts/options</b> default option file directory.</li> <li><b>/var/opt/SUNWvts/logs</b> default log file directory.</li> </ul>
<b>SEE ALSO</b>	<b>sunvts(1M), vtsui(1M), vtsui.ol(1M), vtstty(1M), vtsprobe(1M)</b>



<b>NAME</b>	vtsprobe – prints the device probe information from the SunVTS kernel
<b>SYNOPSIS</b>	<b>vtsprobe</b> [ <b>-m</b> ] [ <b>-h</b> <i>hostname</i> ]
<b>AVAILABILITY</b>	<b>SUNWvts</b>
<b>DESCRIPTION</b>	<b>vtsprobe</b> is a utility that displays the device and configuration information contained in the SunVTS kernel. The output includes the SunVTS assigned group for the device, the device name, the device instance, the testname attached to this device, and the configuration information obtained from the device-specific test probe.
<b>OPTIONS</b>	<p><b>-m</b> Specifies manufacturing mode, which displays the probe information in a format that is easy to read using script files.</p> <p><b>-h</b> <i>hostname</i> Specifies the <i>hostname</i> to connect to and get the device and configuration information. If not specified, the current host will be used.</p>
<b>USAGE</b>	After the SunVTS kernel is up and running, you may type <b>vtsprobe</b> at the shell prompt to get the probe output. (See the <b>sunvts (1M)</b> man page for more information on how to start up SunVTS.
<b>EXAMPLE</b>	<p>Running <b>vtsprobe</b> on a sun4m SPARCclassic produces the following output:</p> <pre> % vtsprobe  Processor(s)   system(system)     System Configuration=sun4m SPARCclassic     System clock frequency=50 MHz     SBUS clock frequency=25 MHz   fpu(fputest)     Architecture=sparc     Type=TI TMS390S10 or TMS390S15 microSPARC chip  Memory   kmem(vmem)     Total: 143120KB   mem(pmem)     Physical Memory size=24 Mb  SCSI-Devices(esp0)   c0t2d0(rawtest)     Capacity: 638.35MB     Controller: esp0     Vendor: MICROP     SUN Id: 1588-15MBSUN0669     Firmware Rev: SN0C </pre>

```
Serial Number: 1588-15MB103
c0t2d0(fstest)
  Controller: esp0
c0t3d0(rawtest)
  Capacity: 404.65MB
  Controller: esp0
  Vendor: SEAGATE
  SUN Id: ST1480 SUN0424
  Firmware Rev: 8628
  Serial Number: 00836508
c0t3d0(fstest)
  Capacity: 404.65MB
  Controller: esp0
  Vendor: SEAGATE
  SUN Id: ST1480 SUN0424
  Firmware Rev: 8628
  Serial Number: 00836508
c0t3d0(fstest)
  Controller: esp0
c0t6d0(cdtest)
  Controller: esp0
tape1(tapetest)
  Drive Type: Exabyte EXB-8500 8mm Helical Scan
Network
  isdn0(isdntest)
    NT Port TE Port
  le0(nettest)
    Host_Name: ctech84
    Host Address: 129.146.210.84
    Host ID: 8001784b
    Domain Name: scsict.Eng.Sun.COM
Comm.Ports
  zs0(sptest)
    Port a -- zs0 /dev/term/a : /devices/ ... a
    Port b -- zs1 /dev/term/b : /devices/ ... b
Graphics
  cgthree0(fbtest)

OtherDevices
  bpp0(bpptest)
    Logical name: bpp0
  sound0(audio)
    Audio Device Type: AMD79C30
  sound1(audio)
    Audio Device Type: DBRI Speakerbox
```

**spd0(spctest)**  
**Logical name: spd0**

**NOTES** The output of **vtsprobe** is highly dependent on the device being correctly configured into the system (so that a SunVTS probe for the device can be run successfully on it) and on the availability of a device-specific test probe.

If the device is improperly configured or if there is no probing function associated with this device, **vtsprobe** cannot print any information associated with it.

**SEE ALSO** **sunvts(1M)**, **vtsk(1M)**, **vtsui(1M)**, **vtsui.ol(1M)**, **vtstty(1M)**

<b>NAME</b>	vtstty – TTY interface for SunVTS																
<b>SYNOPSIS</b>	<b>vtstty</b> [ <b>-qv</b> ] [ <b>-h</b> <i>hostname</i> ]																
<b>AVAILABILITY</b>	<b>SUNWvts</b>																
<b>DESCRIPTION</b>	<b>vtstty</b> is the default interface for SunVTS in the absence of a windowing environment. It can be used in a non-windowing environment such as a terminal connected to the serial port of the system. However, its use is not restricted to this; <b>vtstty</b> can also be used from shell window.																
<b>OPTIONS</b>	<p><b>-q</b>     The "auto-quit" option automatically quits when the conditions for SunVTS to quit are met.</p> <p><b>-v</b>     Prints the <b>vtstty</b> version. The interface is not started when you include this option.</p> <p><b>-h</b> <i>hostname</i>           Connects to the SunVTS kernel running on the host identified by <i>hostname</i>.</p>																
<b>USAGE</b>	<p>The <b>vtstty</b> screen consists of four panels: main control, status, test groups, and console. The panels are used to display choices that the user can select to perform some function and/or to display information. A panel is said to be "in focus" or in a "selected" state when it is surrounded by asterisks and the current item is highlighted. In order to choose from the items in a panel, the focus should be shifted to that panel first.</p> <p>The following are the different types of selection items that can be present in a panel:</p> <table border="0"> <tr> <td style="padding-right: 20px;">Text string</td> <td>Describes a choice that, when selected, either pops up another panel or performs a function. For example, "stop" will stop the SunVTS testing.</td> </tr> <tr> <td>Data entry field</td> <td>To enter or edit numeric or textual data.</td> </tr> <tr> <td>Checkbox</td> <td>Represented as "[ ]". Checkboxes are associated with items and indicate whether the associated item is selected or not. A checkbox can be in one of the following two states: Deselected [ ] or Selected [*].</td> </tr> </table> <p>The key assignments given below describe the keys for shifting focus, making a selection, and performing other functions:</p> <table border="0"> <tr> <td style="padding-right: 20px;">TAB or &lt;CTRL&gt;W</td> <td>Shift focus to another panel</td> </tr> <tr> <td>RETURN</td> <td>Select current item</td> </tr> <tr> <td>Spacebar</td> <td>Toggle checkbox</td> </tr> <tr> <td>Up arrow or &lt;CTRL&gt;U</td> <td>Move up one item</td> </tr> <tr> <td>Down arrow or &lt;CTRL&gt;N</td> <td>Move down one item</td> </tr> </table>	Text string	Describes a choice that, when selected, either pops up another panel or performs a function. For example, "stop" will stop the SunVTS testing.	Data entry field	To enter or edit numeric or textual data.	Checkbox	Represented as "[ ]". Checkboxes are associated with items and indicate whether the associated item is selected or not. A checkbox can be in one of the following two states: Deselected [ ] or Selected [*].	TAB or <CTRL>W	Shift focus to another panel	RETURN	Select current item	Spacebar	Toggle checkbox	Up arrow or <CTRL>U	Move up one item	Down arrow or <CTRL>N	Move down one item
Text string	Describes a choice that, when selected, either pops up another panel or performs a function. For example, "stop" will stop the SunVTS testing.																
Data entry field	To enter or edit numeric or textual data.																
Checkbox	Represented as "[ ]". Checkboxes are associated with items and indicate whether the associated item is selected or not. A checkbox can be in one of the following two states: Deselected [ ] or Selected [*].																
TAB or <CTRL>W	Shift focus to another panel																
RETURN	Select current item																
Spacebar	Toggle checkbox																
Up arrow or <CTRL>U	Move up one item																
Down arrow or <CTRL>N	Move down one item																

Left arrow	or	<CTRL>P	Move left one item
Right arrow	or	<CTRL>R	Move right one item
Backspace			Delete text in a data entry field
ESC			Dismiss a pop-up
<CTRL>F			Scroll forward in a scrollable panel
<CTRL>B			Scroll backward in a scrollable panel
<CTRL>X			Quit <b>vtstty</b> but leave the SunVTS kernel running
<CTRL>L			Refresh the <b>vtstty</b> screen

**NOTES**

1. To run **vtstty** from a telnet session, carry out the following steps:
  - a. Before telnet-ing, determine the values for "rows and "columns". (See **stty(1)** ).
  - b. Set term to the appropriate type after telnet-ing(for example, **set term=vt100** ).
  - c. Set the values of columns and rows to the value noted above. (See **stty(1)** ).
2. Before running **vtstty** ensure that the environment variable describing the terminal type is set correctly.

**SEE ALSO**

**sunvts(1M)**, **vtsk(1M)**, **vtsui(1M)**, **vtsui.ol(1M)**, **vtsprobe(1M)**

<b>NAME</b>	vtsui – SunVTS Graphic User Interface (CDE)
<b>SYNOPSIS</b>	<b>vtsui</b> [ <b>-qv</b> ] [ <b>-h</b> <i>hostname</i> ]
<b>AVAILABILITY</b>	<b>SUNWvts</b>
<b>DESCRIPTION</b>	<p>The <b>vtsui</b> command starts up the CDE Motif version of SunVTS graphic user interface. There can be multiple instances of <b>vtsui</b> running at the same time, all connected to one SunVTS diagnostic kernel, <b>vtsk</b>(1M). The name of the host machine running the diagnostic kernel, <b>vtsk</b>(1M), will be displayed in the title bar of the graphical user interface window.</p> <p><b>vtsui</b> is automatically started up by the <b>sunvts (1M)</b> command. <b>vtsui</b> can be also used to start <b>vtsk (1M)</b> if <b>inetd (1M)</b> is in operation. In that case, the security file, <b>sunvts_sec</b>, will be checked for the permission before running <b>vtsk</b> on the target host. See the "SunVTS User's Guide" for a complete description on using the graphical user interface.</p>
<b>OPTIONS</b>	<p><b>-q</b>      Quits the SunVTS graphic user interface when testing has terminated.</p> <p><b>-v</b>      Displays graphic user interface version information only.</p> <p><b>-h</b> <i>hostname</i>  Starts the SunVTS graphic user interface and connects to the SunVTS diagnostic kernel running on <i>hostname</i>, or invokes the kernel if not running, after security checking succeeds. If <i>hostname</i> not specified, the local host is assumed.</p>
<b>EXIT STATUS</b>	<p>The following exit values are returned:</p> <p><b>0</b>      Successful completion.</p> <p><b>1</b>      An error occurred.</p>
<b>SEE ALSO</b>	<b>sunvts(1M), vtsk(1M), vtsui.ol(1M), vtstty(1M), vtsprobe(1M)</b>

<b>NAME</b>	vtsui.ol – SunVTS Graphic User Interface (OpenLook)
<b>SYNOPSIS</b>	<b>vtsui.ol</b> [ <b>-qv</b> ] [ <b>-h</b> <i>hostname</i> ]
<b>AVAILABILITY</b>	<b>SUNWvts</b>
<b>DESCRIPTION</b>	<p>The <b>vtsui.ol</b> command starts up the OpenLook version of <b>SunVTS</b> graphic user interface. There can be multiple instances of <b>vtsui.ol</b> running at the same time, all connected to one <b>SunVTS</b> diagnostic kernel, <b>vtsk(1M)</b>. The name of the host machine running the diagnostic kernel, <b>vtsk(1M)</b>, will be displayed in the title bar of the graphic user interface window.</p> <p><b>vtsui.ol</b> can be used to start <b>vtsk(1M)</b> if <b>inetd(1M)</b> is in operation. In that case, the security file, <b>.sunvts_sec</b>, will be checked for the permission before running <b>vtsk</b> on the target host. <b>vtsui.ol</b> is also automatically started up by the <b>sunvts(1M)</b> command. See the "SunVTS User's Guide" for a complete description on using the graphic user interface.</p>
<b>OPTIONS</b>	<p><b>-q</b>      Quits the SunVTS graphic user interface when testing has terminated.</p> <p><b>-v</b>      Displays graphic user interface version information only.</p> <p><b>-h</b> <i>hostname</i>  Starts the SunVTS graphic user interface and connects to the <b>SunVTS</b> diagnostic kernel running on <i>hostname</i>, or invokes the kernel if not running, after security checking succeeds. If <i>hostname</i> not specified, the local host is assumed.</p>
<b>EXIT STATUS</b>	<p>The following exit values are returned:</p> <p><b>0</b>      Successful completion.</p> <p><b>1</b>      An error occurred.</p>
<b>SEE ALSO</b>	<b>sunvts(1M), vtsk(1M), vtsui(1M), vtstty(1M), vtsprobe(1M)</b>

**NAME** afb – Elite3D graphics accelerator driver

**DESCRIPTION** **afb** is the device driver for the Sun Elite3D graphics accelerators. The **afbdaemon** process loads the afb microcode at system startup time and during the resume sequence of a suspend-resume cycle.

**FILES** **/dev/fbs/afb*n*** device special file  
**/usr/lib/afb.ucode** afb microcode  
**/usr/sbin/afbdaemon** the afb microcode loader

**SEE ALSO** **afbconfig(1M)**,



# Index

---

## A

afb (7D), 7-26

afbconfig (1M), 1M-1

vtstty (1M), 1M-22

vtsui (1M), 1M-24

vtsui.ol (1M), 1M-25

## D

device and network interfaces, 7-26

## G

GFXconfig (1M), 1M-9

## M

maintenance commands, 1M-1 to 1M-25

## R

rscadm (1M), 1M-13

## S

sunvts (1M), 1M-17

## V

vtsk (1M), 1M-18

vtsprobe (1M), 1M-19

