



SunVTS Reference Manual

Sun Microsystems, Inc.
www.sun.com

Part No. 819-2364-10
January 2006, Revision A

Submit comments about this document at <http://www.sun.com/hwdocs/feedback>

Copyright 2006 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

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, docs.sun.com, Sun StorEdge, Enterprise Network Array, 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.

Federal Acquisitions: Commercial Software—Government Users Subject to Standard License Terms and Conditions.

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 2006 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054 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. La notice suivante est applicable à Netscape Communicator™: (c) Copyright 1995 Netscape Communications Corporation. Tous droits réservés.

Sun, Sun Microsystems, le logo Sun, docs.sun.com, Sun StorEdge, Enterprise Network Array, 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.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFAÇON.



Contents

SunVTS System Administration Commands (1M)

sunvts 1M 1

vts_cmd 1M 3

vtsk 1M 9

vtspoke 1M 11

vtstty 1M 15

vtsui 1M 17

Preface

The *SunVTS Reference Manual* contains reference manual pages (man pages) for SunVTS software provided to Sun hardware customers with the Solaris 10 1/06 product.

How This Book Is Organized

This manual contains the following SunVTS diagnostic software man pages:

- sunvts
- vts_cmd
- vtsk
- vtsprobe
- vtstty
- vtsui

Accessing Sun Documentation Online

You can view, print, or purchase a broad selection of Sun documentation, including localized versions, at:

<http://www.sun.com/documentation>

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can submit your comments by going to:

<http://www.sun.com/hwdocs/feedback>

Please include the part number (819-2364-10) and title of your document with your feedback.

NAME	sunvts - Invokes the SunVTS kernel and its user interface
SYNOPSIS	sunvts [-l <i>epqstv</i>] [-o <i>option_file</i>] [-f <i>log_dir</i>] [-h <i>hostname</i>]
DESCRIPTION	The sunvts 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.
OPTIONS	sunvts supports the following options. -l Displays SunVTS OpenLook graphic interface. -e Disables the security checking feature. -f <i>log_dir</i> Specifies an alternative <i>log_file</i> directory. The default <i>log_file</i> directory is <i>/var/opt/SUNWvts/logs</i>. -h <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. -o <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 <i>/var/opt/SUNWvts/options</i>. -p Starts the SunVTS kernel <i>vt sk(1M)</i> such that it does not probe the test system's devices. -q Automatically quits both the SunVTS kernel and the user interface when testing stops. -s Automatically starts testing from a selected group of tests. The flag must be used with the -o <i>option_file</i> flag. -t Starts <i>vt stty(1M)</i>, a TTY based interface, instead of CDE or OpenLook interface. -v Displays version information from <i>vt sui(1M)</i> and <i>vt sk(1M)</i>.
NOTES	If <i>vt sk(1M)</i> is already running on the test system, the sunvts command ignores the -e , -o , -f , -q , -p , and -s options.
SEE ALSO	<i>vt sk(1M)</i> , <i>vt stty(1M)</i> , <i>vt sui(1M)</i> , <i>vt sui.ol(1M)</i> , <i>vt sprobe(1M)</i>



NAME	vts_cmd - Send a command to the SunVTS kernel (vtsk).	
SYNOPSIS	vts_cmd [command argument]	
DESCRIPTION	vts_cmd is a UNIX shell application that allows you to send a single command to the SunVTS kernel (vtsk). The test machines SunVTS kernel will send the response to the standard output. The SunVTS application programming interface (API) is character based, which means that a string of characters (in the form of a command) can be sent to the SunVTS kernel, which then returns a reply back in the form of a string of characters. vts_cmd(1M) allows the user to send commands and receive replies from a UNIX command line.	
OPTIONS	vts_cmd supports the following options.	
	list testnode	Displays all the testnodes under the specified testnode.
	config testnode	Displays the configuration information of the testnode.
	status [testnode] [-r]	Displays the testing status information of the system. If a testnode is specified, status will display the status information of that testnode. If you use the -r argument, the status information of all of testnodes recursive to the testnode will be displayed.
	option [testnode] [-l] [-h n s t a]	Either displays all the options associated with the specified testnode, or sets a specific option in a testnode. To display a testnodes options, type option followed by the testnode and one of the categories: • -h Threshold • -n Notify category • -s Scheduling category • -t Test execution category • -a Advanced category vts_cmd will print all options, as well as the setting of each option. Use the -l option to display the options in long form. In long form, the options will be displayed with all their settings.

<pre>option [testnode] [test_option] [-g s x y z]</pre>	• -g is used to pass all of the current option settings, for a given instance of a given test, to all of the same instances and tests that are in the same group (will not affect the same tests that are in different groups). • -s is used to pass all of the current option settings for a given instance of a given test, to all of the same instances for all of the same tests on the system (rather than for a group, as with -g). • -x is used to pass all of the current option settings for a given instance of a given test, to all the instances of that test. • -y is used to pass all of the current option settings for a given instance of a given test, to all the instances of all the same tests in a particular group. • -z is used to pass all of the current option settings for a given instance of a given test, to all the instances of all the same tests in the whole system. To set an option, you must state the testnode immediately followed by the option and the new setting. You must use this format when setting an option: <pre>vts_cmd option testnode[option:setting]</pre> Once the option has been successfully changed, vts_cmd will display the word "DONE".
<pre>select testnode</pre>	Selects a testnode. If a testnode is selected, all the tests associated with the testnode will be enabled and run when testing begins. For example, if you select the Graphics testnode, all the tests in Graphics will be enabled for testing. If you select just the "fpu(fputest)" test, then you will only enable this test.
<pre>deselect testnode</pre>	Deselects a testnode. If a testnode is deselected, all the tests associated with the testnode will be disabled and will not be run when testing begins. For example, if you deselect the OtherDevices testnode, all the tests in the OtherDevices will be disabled. If you select just the "cgsix0(cg6)" test, then you will only enable this test.

<code>start</code>	Starts all enabled (selected) SunVTS tests.
<code>stop</code>	Stops all running SunVTS tests.
<code>suspend</code>	Suspends (or pauses) all running SunVTS tests. When you are ready to resume testing, type "resume".
<code>resume</code>	Resumes any suspended tests.
<code>reset</code>	Resets all the SunVTS pass and error counts to zero.
<code>probe</code>	Probes all the devices on the test machine and updates the SunVTS kernels device list. If a device is listed in the device list, but it is not found during the probe, it will be removed from the list. Conversely, if a device does not exist in a previous device list and is found during the probe, it will be added to the list.
<code>load option_file</code>	Loads an option file. Once loaded, the system and test options will be changed to reflect the settings listed in the option file. Option files are stored in the <code>/var/opt/SUNWvts</code> options directory.
<code>store option_file</code>	Creates an option file, listing all the system and test options, and save it in the <code>/var/opt/SUNWvts/options</code> directory.
<code>quit</code>	Terminates the SunVTS kernel (vtsk).
<code>invokeds</code>	Starts the deterministic scheduler.
<code>quitds</code>	Terminates the deterministic scheduler.
<code>loadseq sequence_file</code>	Loads a sequence file. Once loaded, the deterministic scheduler UI will reflect the tasks in the loaded sequence file.
<code>storeseq sequence_file</code>	Creates <code>sequence_file</code> , listing all the tasks in the directory <code>/var/opt/SUNWvts/sequences</code> .
<code>statusseq</code>	Returns a string containing the status information of the currently running sequence. The string consists of four fields separated by commas (", "). The fields are: current status of SunVTS, current loop count of the sequence, total loop count of the sequence, and currently running tasks position.

<code>startseq</code>	Starts the execution of the deterministic scheduler.
<code>stopseq</code>	Stops the execution of the currently running task in the sequence file. Upon starting again, the execution will start from the task that was stopped.
<code>resumeseq</code>	Restarts the execution of the sequence file. Execution will start at the point where the sequence was stopped, unless the sequence was reset, in which case it would start at the beginning of the sequence file.
<code>resetseq</code>	Sets the starting point of the execution to the start of the sequence file. Will also reset the passes and error count.
<code>suspendseq</code>	Suspends the execution of the currently running task in the sequence file.
<code>removeseq sequence_file</code>	Removes <code>sequence_file</code> from the list of sequence files in the directory <code>/var/opt/SUNWvts/sequences</code> .
<code>listtask</code>	Lists the tasks that are present in the currently loaded sequence file.
<code>addtask task_name [i]</code>	Adds <code>task_name</code> at the <code>i</code> th position in the sequence file. If no index is passed then the task would be added to the end of the list.
<code>deletetask [i]</code>	Removes the task at the specified index from the selected sequence.
<code>loadtask task_name</code>	Loads a task file. Once loaded, the system and test options will be changed to reflect the settings listed in the task file.
<code>setloopcount count</code>	Sets the number of loops to run in the current sequence to <code>count</code> .
<code>getvtsmode</code>	Gets the current mode of SunVTS kernel.

EXAMPLES

EXAMPLE 1 To list out the configuration information of the test machine, you would use the `config` command:

```
sample% vts_cmd config /
/[Hostname:sample,Model:SPARCstation-10,SunVTS
version:1.0]:idle
```

EXAMPLE 2 To load an option file, you would use the load command:

```
sample% ls /var/adm/sunvtslog/options
CPU_options sample options
sbus_standard
sample% vts_cmd load sbus_standard
DONE
```

EXAMPLE 3 To print all the system options in the Comm.Ports testnode, you would use the option command and pipe the output to your local printer:

```
sample% vts_cmd option Comm.Ports -l ú lp
request id is printer-213 (standard input)
```

NOTES

VTS_CMD_HOST=hostname – The hostname of the test machine running the SunVTS kernel (vtsk). If this environment variable is not set, vts_cmd will attempt to send the commands to the local machines SunVTS kernel.

SEE ALSO

SunVTS User's Guide



NAME	vtsk - SunVTS diagnostic kernel
SYNOPSIS	vtsk [<i>epqsv</i>] [<i>-o options_file</i>] [<i>-f logfile_directory</i>]
AVAILABILITY	SUNWvts
DESCRIPTION	The vtsk command starts up the SunVTS diagnostic kernel as a background process. There can only be one copy of vtsk running at a time. Only the superuser can execute this command. Normally, vtsk is automatically started up by the sunvts(1M) command if it is not already running. vtsk will also be invoked by inetd(1M) when there is a connection request from vtsui or vtsui.ol. In that case, the security file, .sunvts_sec, will be checked for the permission before running vtsk on the target host specified by vtsui(1M) or vtsui.ol(1M).
OPTIONS	vtsk supports the following options. -e Enables the security checking for all connection requests. -p Starts SunVTS diagnostic kernel, but does not probe system configuration. -q Quits both the SunVTS diagnostic kernel and the attached User Interfaces when the testing is completed. -s Runs enabled tests immediately after started. -v Display SunVTS diagnostic kernels version information only. -o options_file Starts the SunVTS diagnostic kernel and sets the test options according to the option file named options_file. -f logfile_directory Specifies an alternative logfile directory, other than the default.
EXIT STATUS	The following exit values are returned: 0 Successful Completion -1 An error occurred.
NOTES	The vcdiag utility uses the following file: /var/opt/SUNWvts/options default option file directory. /var/opt/SUNWvts/logs default log file directory.
SEE ALSO	sunvts(1M) , vtsui(1M) , vtsui.ol(1M) , vtstty(1M) , vtsprobe(1M)



NAME	<code>vtsp</code> - prints the device probe information from the SunVTS kernel.
SYNOPSIS	<code>vtsp</code> [<i>m</i>] [<i>h</i> <i>hostname</i>]
AVAILABILITY	SUNWvts
DESCRIPTION	<code>vtsp</code> 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.
OPTIONS	<code>vtsp</code> supports the following options. <code>-m</code> Specifies manufacturing mode, which displays the probe information in a format that is easy to read using script files. <code>-h <i>hostname</i></code> Specifies the hostname to connect to and get the device and configuration information. If not specified, the current host will be used.
EXAMPLES	EXAMPLE 4 Running <code>vtsp</code> on a sun4m SPARCclassic produces the following output: <pre>% vtsp Processor(s) system(systest) 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</pre>

```
SCSI-Devices(esp0)
  c0t2d0(rawtest)
    Capacity: 638.35MB
    Controller: esp0
    Vendor: MICROP
    SUN Id: 1588-15MBSUN0669
    Firmware Rev: SN0C
    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(spptest)``Logical name: spd0`**NOTES**

The output of `vtspoke` 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, `vtspoke` cannot print any information associated with it.

SEE ALSO

`sunvts(1M)`, `vtstk(1M)`, `vt sui(1M)`, `vt sui.ol(1M)`, `vtstty(1M)`



NAME	<code>vtstty</code> - TTY interface for SunVTS
SYNOPSIS	<code>vtstty [qv -h <i>hostname</i>]</code>
DESCRIPTION	<code>vtstty</code> 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; <code>vtstty</code> can also be used from shell window.
OPTIONS	<code>vtstty</code> supports the following options. -q The "auto-quit" option automatically quits when the conditions for SunVTS to quit are met. -v Prints the <code>vtstty</code> version. The interface is not started when you include this option. -h <i>hostname</i> Connects to the SunVTS kernel running on the host identified by <i>hostname</i>.
USAGE	The <code>vtstty</code> 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. The following are the different types of selection items that can be present in a panel: - 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 [X]. The key assignments given below describe the keys for shifting focus, making a selection, and performing other functions: - 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 vtstty but leave the SunVTS kernel running
<CTRL>L	Refresh the vtstty 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)

NAME	vtstui - SunVTS Graphic User Interface (CDE)
SYNOPSIS	vtstui [-qz -h <i>hostname</i>]
DESCRIPTION	vtstui command starts up the CDE Motif version of SunVTS graphic user interface. There can be multiple instances of vtstui running at the same time, all connected to one SunVTS diagnostic kernel, vtsk(1M). The name of the host machine running the diagnostic kernel, vtsk(1M), will be displayed in the title bar of the graphical user interface window. vtstui is automatically started up by the sunvts(1M) command. vtstui can be also used to start vtsk(1M) if inetd(1M) is in operation. In that case, the security file, sunvts_sec, will be checked for the permission before running vtsk on the target host. See the "SunVTS Users Guide" for a complete description on using the graphical user interface.
OPTIONS	vtstui supports the following options. -q Quits the SunVTS graphic user interface when testing has terminated. -v Displays graphic user interface version information only. -h <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.
EXIT STATUS	The following exit values are returned: 0 Successful Completion 1 An error occurred.
SEE ALSO	sunvts(1M) , vtsk(1M) , vtstui.ol(1M) , vtstty(1M) , vtspoke(1M)

