

SunVTS 7.0 Software

User's Guide for Initial Release Through Patch Set 7



Copyright © 2007, 2010, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related software documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Copyright © 2007, 2010, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quel que procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. UNIX est une marque déposée concédée sous licence par X/Open Company, Ltd.



Contents

Preface xv

1. SunVTS 7.0 Software Overview 1

SunVTS Introduction 1

Terminology 2

Functional Architecture 2

Testing With SunVTS 7.0 Software 4

Test Modes 4

Online Mode 5

System Exerciser 5

Component Stress 6

Test Options 6

User Interfaces 7

Graphical User Interface 7

Browser User Interface 7

Terminal User Interface 7

Command-Line Interface 8

Harness and Middle Server Functions 8

Harness 8

The Middle Server 8

2.	SunVTS 7.0 Packages and Running Instructions	9
	SunVTS 7.0 Packages	9
	Installation Requirements	11
	SunVTS 7.0 Man Pages	11
	▼ To Set Up Access to the SunVTS Man Pages	11
	Requirements for Running SunVTS	12
	Runtime Considerations	12
	SunVTS Test Procedures Overview	13
	Preparing Devices for Testing	14
	Starting SunVTS	14
	▼ To Start SunVTS	15
	Starting a Specific User Interface	15
	▼ To Start the BUI on the Same System as the Host Under Test	15
	▼ To Start the BUI on a System Other Than the Host Under Test	15
	▼ To Start the Graphical User Interface on the Same System as the Host Under Test	16
	▼ To Start the TTY User Interface on the Same System as the Host Under Test	17
	▼ To Start the Command-Line Interface on the Same System as the Host Under Test	17
	Help Options	18
	Interpreting SunVTS Messages	18
	SunVTS on LDoms Enabled Systems	21
	Backward Compatibility	22
	Displaying Devices Using the VTSpoke Utility	22
3.	Using the SunVTS Browser User Interface	25
	Starting the Browser User Interface	25
	▼ To Start the SunVTS Browser User Interface	26
	Host Machine View	28

▼ To Add a Host	29
▼ To Remove a Host	32
▼ To View Host Configuration	33
Test Group Screen	35
Test Group	36
Test Table Contents	38
Test Group Button Functions	39
Test Modes	40
Test Session	40
▼ To Save a Session	40
▼ To List a Session	42
▼ To Load a Session	43
▼ To Delete a Session	44
Reset Session	45
Test Options Screen	45
▼ To Change Test Options	48
▼ To Set Global Options	50
Safe and Unsafe Options	51
▼ To Create a Configuration File for the Unsafe Option	51
Logs Screen	52
Types of Logs	54
Test Error	54
vtsk Error	54
Test Information	54
Unix Messages	54
Test Verbose Messages	54
Managing Logs	55
View Logs	55

Delete Logs 55

4. Using the Graphical User Interface 57

Starting the Graphical User Interface 57

Using the Test Group Screen 58

Test Group Table 58

Test Table Contents 59

Test Group Button Functions 61

▼ To View the Host Configuration 61

Test Modes 62

Test Session 62

Save Session 63

▼ To Save a Test Session 63

▼ To List a Session 64

▼ To Load a Session 65

▼ To Delete a Session 66

Reset Session 67

Test Options Screen 67

▼ To Change Test Options 68

▼ To Set Global Options 69

Safe and Unsafe Options 71

▼ To Create a Configuration File for the Unsafe Option 71

Using the Logs Screen 72

Types of Logs 72

Test Error 72

Vtsk Error 73

Test Information 73

Unix Messages 73

Test Verbose Messages 73

Managing Logs	73
View Logs	73
Delete Logs	74
Using the Auto Mode Sequencer	74
▼ To Edit Auto Mode Sequencer Options	74
5. Using the SunVTS TTY User Interface	77
Starting the SunVTS TTY User Interface	77
Navigating the SunVTS TTY User Interface	78
Configuring SunVTS for a Test Session	80
Identifying the SunVTS Status	80
Selecting a Test Mode	81
Changing the Global Options	81
Changing Test Specific Options	83
Enabling or Disabling Tests	86
Running and Monitoring a Test Session	86
Starting the Test Session	86
Monitoring Tests	86
Stopping the Test Session	87
Reviewing and Resetting Test Session Results	87
Reviewing Logs	87
Resetting Test Session Results	89
Additional SunVTS TTY User Interface Features	89
Saving a Test Session Configuration for Repeated Use (Session Files)	90
▼ To Create a Session File	90
▼ To Load a Session File	91
▼ To Remove a Session File	92
Viewing Host System Configuration	92
Reprobing the Tests	93

Displaying SunVTS TTY UI Help and Release Table	94
Using the Auto Mode Sequencer	95
▼ To Edit Auto Mode Sequencer Options	95
Quitting the TTY User Interface	96

6. Using the SunVTS Command-Line Interface 99

Command-Line Interface Overview	99
---------------------------------	----

Command Options	100
-----------------	-----

get_agent_summary	100
get_host_config	100
get_status <i>test-name</i>	101
list_tests	101
enable_test <i>test-name</i> /all	101
disable_test <i>test-name</i> /all	102
get_mode	102
set_mode [<i>mode</i>]	102
get_global_options	103
set_global_options [<i>Duration of Testing</i> :<0-99999 mins>, <i>Verbose</i> :Enable/Disable]	103
get_test_options <i>test-name</i>	104
set_test_options <i>test-name</i> [<i>Stress</i> :low/high, <i>Scheduling Policy</i> :Test Time<mins>/Test Passes, Test Time<mins>:x*, Test Passes:y*, Error Limit:z*]	104
get_device_options [<i>testname.devicename</i>]	105
set_device_options [<i>testname.devicename</i> : <i>Exported Variable1</i> :Value1, <i>Exported Variable2</i> :Value 2,...]	105
save_session [<i>overwrite</i> :yes/no, <i>defaultMode</i> :test mode, <i>name</i> :session name]	106
list_sessions	106
load_session[<i>defaultMode</i> :testMode, <i>name</i> :sessionName]	106
get_session_name	107

- delete_session *sessionName* 107
- reset_session 108
- start 108
- stop 108
- reset 108
- reprobe 109
- get_modesequencer_options 109
- set_modesequencer_options 110
- quit 111

7. SunVTS 7.0 Tests 113

- Test Requirements 113

- Disk Test 114

- Description 114

- Disk Subtests 114

- Test Modes 115

- Test Setup 115

- Test Options 115

- Removable Disk Test 115

- Media subtest 116

- File System Subtest 116

- Test Modes 116

- Environment Test 117

- Test Requirements 118

- Test Modes 118

- Graphics Test 119

- Test Modes 119

- Test Setup 119

- ▼ To Disable the Solaris Screen Saver 119

▼ To Turn Display Power Management Off	120
▼ To Change the Screen Resolution	120
Host Bus Adapters Test	120
Test Modes	121
Test Setup	121
Interconnect Test	121
Test Modes	122
Test Setup	122
IOports Test	122
Test Modes	123
Test Setup	123
Media Test	123
Test Modes	124
Test Setup	124
Memory Test	124
Swap Space Requirements	125
Test Modes	125
Test Options	125
Network Test	125
Test Modes	126
Test Setup	126
Test Options	126
Network Tests Options	127
▼ To Select and Deselect Network Tests	127
Target Options	128
Loopback Options	128
▼ To Choose Auto, Internal, or External Loopback	128
Processor Test	129

Test Modes 130

8. SunVTS Bootable CD and USB Environment for Sun x86 Platforms 131

Bootable CD and USB Environment Introduction 131

Using Bootable CD or USB Media to Start SunVTS 132

▼ To Boot Using the CD Using TTY, BUI, or GUI 132

▼ To Boot Using the USB 133

Using the Service Processor to Boot from the SunVTS Image (Booting from a Virtual Machine) 134

▼ To Do the Boot Process 134

Limitations 136

▼ To Obtain a Complete FMA Report 136

A. Connectors 137

Connector Overview 138

25-Pin RS-232 Plug 139

25-Pin RS-232 Port-to-Port Cable 141

8-Pin to 8-Pin Cable 143

8-Pin Plug 144

25-Pin Port A-to-Port B Plug 145

25-Pin Port A-to-A Port B-to-B
Plug 147

96-Pin Female Connector 149

96-Pin Female Special Connector 151

37-Pin RS-449 Cable 152

37-Pin RS-449 Plug 153

9-Pin Male Single-Port Plug 154

9-Pin Female Single-Port Plug 154

9-Pin to 25-Pin Port-to-Port Cable 155

9-Pin to 9-Pin Port-to-Port Cable 156

NT to TE Cable	156
Twisted-Pair Ethernet (TPE) Cable for Fast Ethernet	157
TPE Cable for Gigabit and 10/100 Ethernet	157
9-Pin Male Single-Port Plug	159
9-Pin Female Single-Port Plug	159
9-Pin Male DB-9 External Connector	160
9-Pin Female DB-9 External Connector	161

B. Frequently Asked Questions (FAQ) and General Recommendations on SunVTS Usage 163

Introduction	163
Frequently Asked Questions (FAQ)	164
When Should I Use the SunVTS Tool?	164
What Do the Different Modes and the High/Low Within the Modes Mean?	
When Should I Use Them?	165
How Long Should I Run the Tests?	166
The Test Did Not Fail. Is Everything Ok with My Hardware?	167
Your Tool Caused My System to Panic. Are You Doing Anything that a Normal Application Would Not Do?	168
SunVTS Logs Are Showing a Failure, and the Cause Is Being Attributed to the System Running Out of Resources While Running Tests. Is This a Real Error in My System?	168
I See that the New SunVTS 7.0 Version Is Very Different from the Version I Have Been Using. What Are the Main Differences?	169
I Have Been Using the Previous Generation of SunVTS. I Used to Create Option Files for My Options and Load Them. Can I Use the Same Option Files With the New SunVTS 7.0 Version of the Tool?	169
What Is the Meaning of Improved Diagnostics Effectiveness?	170
Who Do I Contact if I Have More Questions?	170

C. Utilities 171

VTsreportgenerate Utility	171
---------------------------	-----

Syntax for VTSreportgenerate	172
SunVTS Summary Test Report	172
SunVTS Test Report Header	172
Test Status	172
Faults and Suspects	173
High Priority SunVTS Messages	173
syslog(3C) Messages	174
SunVTS Verbose Test Report	174
Test Status	175
Faults and Suspects	176
SunVTS Log Messages	177
syslog(3C) Messages	177
Index	179

Preface

This guide describes how to install, configure, and use Oracle's Sun Validation Test Suite (SunVTS) software from the initial release through patch set 7. The primary audience of this manual is hardware testing and verification personnel, qualified service-trained maintenance providers, and advanced system end users.

The SunVTS software provides a comprehensive diagnostic tool that tests and validates Sun hardware by verifying the connectivity and proper functioning of most hardware controllers and devices on Sun platforms.

Note – The Solaris Operating System (OS) release with which this version of the SunVTS software is delivered supports all Sun manufactured SPARC and X86 systems, running the Solaris operating system.

In this document the term x86 refers to 64-bit and 32-bit systems manufactured using processors compatible with the AMD64 or Intel Xeon/Pentium product families.

Using UNIX Commands

This document might not contain information about basic UNIX commands and procedures such as shutting down the system, booting the system, and configuring devices. Refer to the following for this information:

- Software documentation that you received with your system
- Solaris Operating System documentation, which is at:

<http://docs.sun.com/app/docs/prod/solaris#hic>

Shell Prompts

Shell	Prompt
C shell	<i>machine-name%</i>
C shell superuser	<i>machine-name#</i>
Bourne shell and Korn shell	\$
Bourne shell and Korn shell superuser	#

Typographic Conventions

Typeface*	Meaning	Examples
AaBbCc123	The names of commands, files, and directories – on-screen computer output	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. % You have mail.
AaBbCc123	What you type, when contrasted with on-screen computer output	% su Password:
<i>AaBbCc123</i>	Book titles, new words or terms, words to be emphasized. Replace command-line variables with real names or values.	These are called <i>class</i> options. You <i>must</i> be superuser to do this. To delete a file, type <code>rm filename</code> .

* The settings on your browser might differ from these settings.

Related Documentation

The following table lists the documentation for this product. The online documentation is available at:

<http://docs.sun.com/app/docs/prod/test.validate>

Application	Title	Part Number	Format
Late breaking information	See the Software Release Notes for each release.	Multiple	PDF HTML
Using the initial release of SunVTS software through patch set 7	<i>SunVTS 7.0 Software User's Guide for Initial Release Through Patch Set 7</i>	820-0012-20	PDF HTML
Using the Patch Set 8 and Subsequent compatible releases of SunVTS software.	<i>SunVTS 7.0 Software User's Guide</i>	821-1360	PDF HTML

Documentation, Support, and Training

Sun Function	URL
Documentation	http://www.sun.com/documentation/
Support	http://www.sun.com/support/
Training	http://www.sun.com/training/

Third-Party Web Sites

Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

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://docs.sun.com>

Please include the title and part number of your document with your feedback:
SunVTS 7.0 Software User's Guide for Initial Release Through Patch Set 7, part number 820-0012-20.

SunVTS 7.0 Software Overview

This chapter provides an overview of the SunVTS 7.0 software. Topics include:

- [“SunVTS Introduction” on page 1](#)
- [“Terminology” on page 2](#)
- [“Functional Architecture” on page 2](#)
- [“Testing With SunVTS 7.0 Software” on page 4](#)
- [“User Interfaces” on page 7](#)
- [“Harness and Middle Server Functions” on page 8](#)

SunVTS Introduction

Sun Validation Test Suite (SunVTS) is a Sun hardware validation test based on the Solaris operating system. The suite’s multiple hardware diagnostic tests verify the functionality of most hardware controllers and devices for SPARC and x86 architecture based systems. SunVTS also provides an infrastructure for developers to develop their own tests and integrate into SunVTS.

SunVTS supports execution on a full range of Sun systems, starting from small desktops to high-end servers. The tests and the infrastructure of the tool scales with the type and size of the system configuration. SunVTS automatically detects the type and configuration of the system it is running on and tailors test execution accordingly. The tool can be used for system or component validation during development, production, receiving inspection, troubleshooting, periodic maintenance, and general system exercising.

Terminology

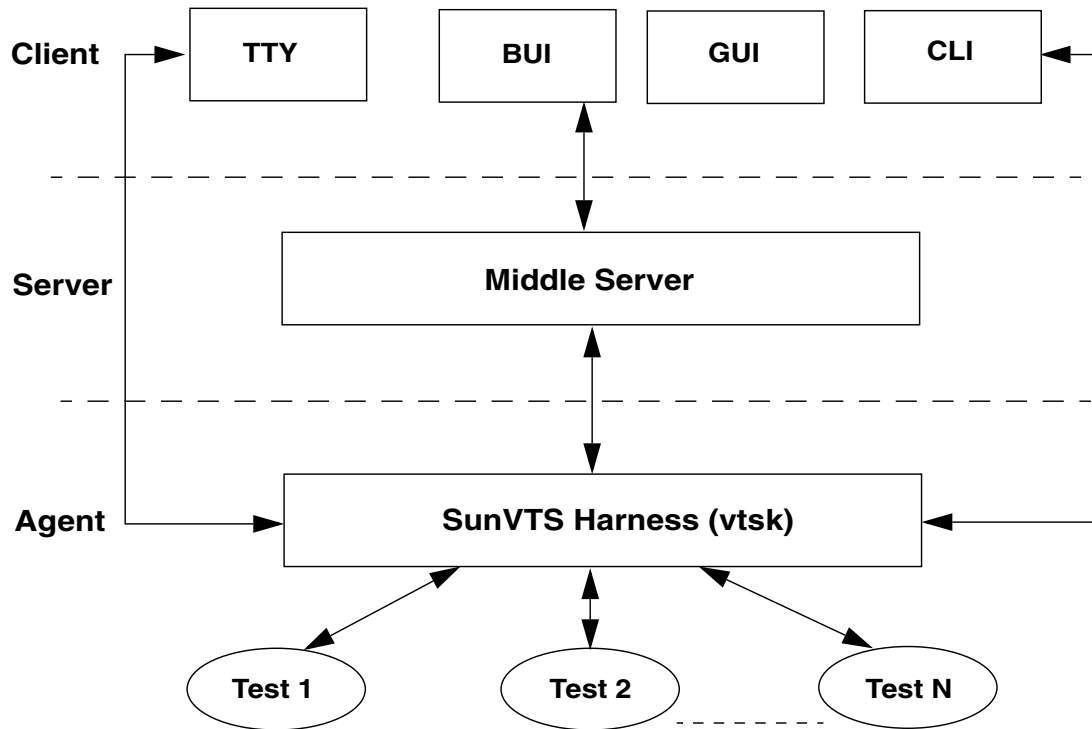
- **SunVTS** – Sun Validation Test Suite
- **BUI** – Browser user interface
- **VTSMS** – Validation test suite middle server
- **vtsk (or harness)** – SunVTS kernel
- **vtskJB** – vtskJB is part of vtsk that enables vtsk to communicate with the BUI

Functional Architecture

SunVTS 7.0 software follows a conventional 3-tier architecture model. It consists of three distinct layers, client, server and agent. [FIGURE 1-1](#) shows the main functional blocks of the tool and how they interact with each other.

- **Client Layer** – Consists of the user interfaces for the SunVTS tool. Three different user interfaces are supported by SunVTS.
 - Graphical User Interface (GUI)
 - Browser user interface (BUI)
 - Terminal user interface (TTY)
 - Command-line interface (CLI)
- **Server layer** – A Java-based program that acts as a server for the BUI. The main purpose of the server layer is to provide facility to the user where they can control test execution on multiple operating system domains, each running a separate SunVTS Agent.

FIGURE 1-1 SunVTS 7.0 Architecture



- **Agent layer** – This layer consists of all the tests for testing the hardware and a harness, which manages the tests. The harness also handles chores like logging or messages, handling of commands coming from the different user interfaces, status updates, and hardware probing.

You run the components in the agent layer on the system under test. The TTY and CLI user interfaces also have to be local on the machine under test. The middle server and BUI can reside on the same machine under test or any other machine on the same subnet. For using the TTY or CLI, you first need to log onto the system under test and then use the interface.

Testing With SunVTS 7.0 Software

The SunVTS 7.0 software is an application level tool that is used for hardware testing and validation during development, production, receiving inspection, troubleshooting, periodic maintenance, and general system exercising. Testing capabilities are provided for all the major hardware in the system. You can use any of the four user interfaces (see [“User Interfaces” on page 7](#)).

Test coverage is provided for most of the key components in the system. The SunVTS software provides a simple and user-friendly set of tests and test options, which can be catered towards different testing needs. The tests that you can use to test your system include the following (see [Chapter 7, “SunVTS 7.0 Tests”](#) on [page 113](#) for test descriptions).

- Processor
- Memory
- Disk
- Removable Disk
- Graphics
- Media
- Ioport
- Interconnects
- Network
- Environment
- HBA

Test Modes

The SunVTS tests can be tailored for different testing needs through a set of modes and test options. These modes and test options determine how the tests are run on the system. You have the option of running the tests in three different modes:

- Online Mode
- System Exerciser
- Component Stress

Each of the modes support two levels of testing, low and high. These two levels of testing determine the degree of intensity of the test in terms of the resource usage and level of stress.

The amount of memory in a system is an important factor in setting the low or high test levels. If the system has a small memory configuration, the default stress level is set to low in each of the modes. A larger memory configuration results in a default stress level that is set to high.

SunVTS considers a system to be a small memory configuration if it has less than or equal to 2 GB of memory for an x86 system and less than or equal to 4 GB of memory for a SPARC system. If the unit under test has less than 2 GB of memory for x86 systems, and less than 4 GB of memory for SPARC systems, you should select the low level.

Note – If you select high on any smaller memory configurations, the tests may exit or run ineffectively due to resource constraints.

Online Mode

Use this mode for testing if you want to run other applications in addition to SunVTS simultaneously on the system. The goal of this mode is to validate hardware without having significant impact on the running of any other applications running on the system. In this mode the tests would not stress on the system and to provide coverage they might have to run for longer periods of time. If there are any latent faults in the system hardware, the tests in this level will try to stimulate those faults before the application is impacted by the fault. When the test stimulate the fault, the OS detects the fault and takes appropriate action.

The following are test characteristics in this mode:

- **Non-Intrusive** – The tests running in this mode do not intrude on the applications that are running on the system. Nonintrusive mode does not change any system configuration that can have an effect on the applications.
- **Data Safe** – The tests do not cause any kind of data corruption.
- **Test time** – The time of completion for these tests could vary from milliseconds to days. There are no restrictions on the amount of time.
- **Resource Usage** – The resource usage is designed so that the test does not have any distinguishable effect on the applications that are running on the system. The goal is to use as few resources as possible for a given instance of time.

System Exerciser

This mode provides system exercising capability. It will test the system with all its components. All tests are run in parallel. In this mode no other application (other than SunVTS) should be running on the system. The following are characteristics of the tests in this mode:

- The tests can be intrusive in nature. No other application should be running on the system while these tests are running.
- The tests are data safe by default.
- The tests are aware that there could be other tests running. Accordingly, the tests would then allocate and lock resources.

Component Stress

This mode performs targeted testing of system components and modules. In this mode the test puts the maximum stress on the component or module under test. The tests are not run simultaneously with other tests: all tests are run sequentially. No other user application should be running on the system. The following are characteristics of the tests in this mode:

- The tests can be intrusive in nature. The tests might need specific system configuration settings to run most efficiently.
- The tests are data safe by default.
- The tests assume that the complete system resources are at their disposal and can use resources according to its needs.

Test Options

In addition to the test modes, SunVTS 7.0 provides intuitive options for running the tests. They can be generic options, that are applicable to all tests, as well as test-specific options that only apply to a particular test.

For example, a generic option is to run the tests as time-based or pass-based, which can be applied to all tests. Another example of a test-specific option would be the Target option in the Network test, where you can specify the IP address of the target system in the network.

There are also global-level options that apply to a testing session. For example, you can specify a time limit for a testing session.

User Interfaces

SunVTS 7.0 software provides four different user interfaces. These user interfaces cater to different usage needs of the tool. Testing session controls are available through all interfaces. The harness, which handles the commands that are coming from the user interfaces, does not distinguish between which interface is sending the command. Below are the different user interfaces and their usages.

Graphical User Interface

The SunVTS graphical user interface (GUI) is a Java-based standalone user interface. The Java GUI needs to be started on the same machine under test that has the SunVTS kernel is running. The GUI does not allow remote connection. See [“Using the Graphical User Interface” on page 57](#).

Browser User Interface

SunVTS browser user interface (BUI) is Java-based and is a part of the Sun Java Web Console. The Java Web Console provides a common point for Sun web-based system management applications to be registered and accessed. The SunVTS BUI can manage multiple testing sessions running on multiple domains. The other two interfaces (CLI and TTY) do not have this capability. The BUI may not be supported in future releases of SunVTS. See [Chapter 3, “Using the SunVTS Browser User Interface” on page 25](#).

Terminal User Interface

The terminal user interface is an ASCII-based screen oriented interface that does not need a graphical window environment. You can use this interface to run SunVTS from a serial terminal or a graphical window tool like shell tool or terminal window. See [Chapter 5, “Using the SunVTS TTY User Interface” on page 77](#).

Command-Line Interface

The command-line interface (CLI) is the most lightweight user interface. Its main advantage is that it can be used in scripts for tool automation. The CLI is provided through the interface called `vtscmd`. It is a UNIX shell application that enables you to send a single command to the SunVTS kernel (`vtsk`), and receive a reply back. See [Chapter 6, “Using the SunVTS Command-Line Interface” on page 99](#).

Harness and Middle Server Functions

Harness

The harness (`vtstk`) acts as the control tower for a testing session. The harness is the entity that controls and manages the complete working of a testing session. This includes functions like starting and stopping tests, message logging, handling commands coming from the user interfaces, scheduling and regular status updates. Like the test, the `vtstk` process needs to be run on the machine under test.

The Middle Server

The SunVTS middle server (`vtSMS`) only comes into use when you use the BUI. The middle server acts as a single point of contact for any Web user interface to communicate with the harness. The middle server provides a uniform API that enables the SunVTS BUI, or any other application, to interact with it. The `vtSMS` takes commands from the BUI, translates them, and sends them to the harness and receives replies. The middle server allows multiple user interfaces to be connected to it. The middle server sends commands to the harness one at a time, and sends a command only after the previous command has completed.

SunVTS 7.0 Packages and Running Instructions

This chapter describes the SunVTS 7.0 software packages, and installation and running of the software. Topics include:

- “SunVTS 7.0 Packages” on page 9
- “Installation Requirements” on page 11
- “SunVTS 7.0 Man Pages” on page 11
- “Requirements for Running SunVTS” on page 12
- “Runtime Considerations” on page 12
- “SunVTS Test Procedures Overview” on page 13
- “Starting SunVTS” on page 14
- “Interpreting SunVTS Messages” on page 18
- “SunVTS on LDom Enabled Systems” on page 21
- “Backward Compatibility” on page 22
- “Displaying Devices Using the `VTSprouce` Utility” on page 22

SunVTS 7.0 Packages

The SunVTS packages are delivered on the Solaris DVD.

SunVTS software is installed by default when the operating system is loaded onto the system. To find and upgrade to the latest version of the SunVTS software, refer to the SunVTS documentation at:

<http://docs.sun.com/app/docs/prod/test.validate>

TABLE 2-1 lists the SunVTS packages that contain the infrastructure and test binaries.

TABLE 2-1 SunVTS 7.0 Software Packages

Package name	Description
SUNWvts	Contains the test development library APIs and SunVTS kernel. You must install this package to run the SunVTS software.
SUNWvtsmn	Contains the man pages for the SunVTS utilities, including the command-line utility.
SUNWvtshr	Contains the SunVTS framework configuration files in the root partition (superuser)
SUNWvtss	Contains the server and browser user interface (BUI) components that must reside on the server system.
SUNWvtsts	Contains the test binaries for both x86 and SPARC systems.

Use the following command to display SunVTS package information:

```
# pkginfo -l SUNWvts SUNWvtsmn SUNWvtshr SUNWvtss SUNWvtsts
```

You can also use either of the following commands to display additional SunVTS package information:

```
# pkginfo | grep vts
```

Or,

```
# showrev -p | grep vts
```

Use either of the following two methods to display SunVTS version information:

```
# cat /usr/sunvts/bin/.version
7.0ps6build5
```

Or,

```
# cd /usr/sunvts/bin
# ./vts_cmd get_version
7.0ps6build5
```

Installation Requirements

The Solaris 10 5/08 or later, or an OpenSolaris build released after Solaris 10 5/08 must be installed (with the Entire Distribution software group [SUNWCa11] at minimum), and booted to the multiuser level (run level 3). You must be superuser (root) to run the tests.

SunVTS 7.0 Man Pages

The SunVTS man pages are installed in the `/usr/sunvts/man` directory. To access the man pages, add this directory to your `MANPATH` variable in the initialization that corresponds to your login shell. (usually, `.profile` for the Bourne and Korn shells or `.login` for the C shell).

▼ To Set Up Access to the SunVTS Man Pages

1. **Using an editor, add the SunVTS man directory (`/usr/sunvts/man`) to the `MANPATH` variable in the appropriate initialization file.**

Bourne, Korn shell example:

```
export MANPATH=/usr/share/man:/usr/man:/usr/sunvts/man:$MANPATH
```

C shell example:

```
setenv MANPATH /usr/share/man:/usr/man:/usr/sunvts/man
```

2. **Source the modified initialization file (with the `.[dot]` or the `source` command), or log out and log back in.**
3. **Verify that the SunVTS man directory is part of the `MANPATH` variable:**

```
# echo $MANPATH
/usr/share/man:/usr/man:/usr/sunvts/man
```

Note – For more information about customizing a user work environment, shell variables, and the initialization files, refer to your Solaris system administration guides.

Requirements for Running SunVTS

The following are required to run SunVTS 7.0:

- The system must be booted to the multiuser level.
- SunVTS software needs Java Web Console in order to use the Browser User Interface (BUI). If your Solaris installation does not have the Java Web Console packages installed, you can obtain the packages from:

<http://www.sun.com/download/products.xml?id=46cb72a9>

Minimum supported browsers include:

- Mozilla 1.4
- Netscape 6.2
- FireFox 1.0

Runtime Considerations

Before you start SunVTS software, consider the following runtime issues:

- **System load** – Before you run SunVTS software on your system, consider the added system activity created by SunVTS software and the effect of that activity on users. Depending on the mode that you select, testing can add significantly to your system load or be quite minimal.
- **Other applications** – Some SunVTS tests stress the system resources (such as virtual memory) to the point that other applications might not run.
- **System functionality** – SunVTS software is a diagnostic tool. SunVTS software runs only when your system is fully booted to run level 3 (multiuser level). SunVTS software will not run from single-user (maintenance mode) nor from run level 0 (monitor mode).
- **Superuser access** – For security reasons, you must be superuser to run SunVTS software.

- **Graphics tests** – When running tests on frame buffers, you must not run any other application or screen saver program that uses the frame buffer. Frame buffer graphic tests print test patterns on the display. This situation temporarily prevents you from working in your window environment.
- **Swap space** – The amount of swap space required for testing varies widely with individual hardware and software configurations. If an adequate amount of swap space is not available on your system, the status column on the browser user interface displays the message waiting for swap space.

SunVTS Test Procedures Overview

[TABLE 2-2](#) describes the process of testing your system with SunVTS software. The table also provides a brief description and reference to required information.

TABLE 2-2 SunVTS 7.0 Test Procedure Overview

Step	Procedures	Description
1	Become superuser (root).	Log in as <code>root</code> or use <code>su</code> to become superuser.
2	Determine the current state of the system you plan to test.	Your system is considered online if other applications are running and the system is operating in a production capacity. If the system is online, you should stop the applications and ensure that the system remains offline for the duration of your testing.
3	Start SunVTS	Refer to “Starting SunVTS” on page 14 .
4	Prepare the devices for testing.	Load the required test media and loopback connectors (see “SunVTS 7.0 Tests” on page 113). Test media is required when testing your CD- or DVD-ROM, and diskette drives.
5	Start the User Interface.	Refer to “Starting the Browser User Interface” on page 25 . Refer to “Starting the SunVTS TTY User Interface” on page 77 .
6	Start the SunVTS software.	For browser-based interface, click the SunVTS link under the systems category in the Java Web Console and add the host that you want to start testing. For the terminal-based user interface, see “Starting the SunVTS TTY User Interface” on page 77 .

TABLE 2-2 SunVTS 7.0 Test Procedure Overview (*Continued*)

Step	Procedures	Description
7	Set up SunVTS with desired tests selection.	Select the desired tests and click Start.
8	Review the test results.	Review test results by opening the log files in the system under test located in <code>/var/sunvts/logs</code> .
9	Quit SunVTS.	Stop all the tests on the system by clicking the Stop button and the Logout button.

Preparing Devices for Testing

Several tests require the installation of media or loopback connectors. Complete such requirements before the SunVTS kernel probes for devices (for example, when SunVTS software is started, or when you run the SunVTS `reprobe` command. This installation allows the SunVTS kernel to properly identify each device.

You must install media in these devices if you plan to test them in the System Exerciser:

- **CD-ROM and DVD drives** – Load appropriate media into the drive. The media can be an audio or data CD.
- **Diskette drive** – Load a scratch diskette in the drive. The diskette test has a read-only mode, but use a scratch tape to avoid inadvertently writing over data.
- **Smart card reader** – Allocate a scratch smart card for testing purposes because the SmartCard test writes on the card.
- **Communication ports** – Many of these tests require a loopback connector attached to the port. Attach any required loopback connectors for the ports you plan to test. For more information about connectors, see [Appendix A](#).
- **Printer ports** – Requires connection to a printer.

Starting SunVTS

You must start SunVTS before selecting a specific user interface: BUI ([Chapter 3](#)), TTY ([Chapter 5](#)) and CLI ([Chapter 6](#)). (See “Starting a Specific User Interface” on [page 15](#).)

▼ To Start SunVTS

- Type the following:

```
# cd /usr/sunvts/bin  
# ./startsunvts
```

This begins an interactive session. SunVTS starts with the selected user interface.

Starting a Specific User Interface

The following describes how to start SunVTS from a specific user interface.

▼ To Start the BUI on the Same System as the Host Under Test

- Type the following:

```
# cd /usr/sunvts/bin  
# ./startsunvts -b
```

Or,

```
# cd /usr/sunvts/bin  
# ./startsunvts --bui
```

▼ To Start the BUI on a System Other Than the Host Under Test

Two steps are involved in starting the BUI on a system other than the host under test:

1. To start the agent and JavaBridge on the system under test, type:

```
# cd /usr/sunvts/bin  
# ./startsunvts -a
```

Or,

```
# cd /usr/sunvts/bin  
# ./startsunvts --agent
```

2. To bring up the BUI on a different system, type:

```
# cd /usr/sunvts/bin  
# ./startsunvts -s
```

Or,

```
# cd /usr/sunvts/bin  
# ./startsunvts --server
```

▼ To Start the Graphical User Interface on the Same System as the Host Under Test

To start the Graphical User Interface (GUI) on the same system as the host under test, type:

```
# cd /usr/sunvts/bin  
# ./startsunvts -g
```

Or,

```
# cd /usr/sunvts/bin  
# ./startsunvts --gui
```

▼ To Start the TTY User Interface on the Same System as the Host Under Test

- Type the following:

```
# cd /usr/sunvts/bin  
# ./startsunvts -t
```

Or,

```
# cd /usr/sunvts/bin  
# ./startsunvts --tty
```

▼ To Start the Command-Line Interface on the Same System as the Host Under Test

- Type the following:

```
# cd /usr/sunvts/bin  
# ./startsunvts -c
```

Or,

```
# cd /usr/sunvts/bin  
# ./startsunvts --commandline
```

Help Options

- To list Help options, type:

```
# cd /usr/sunvts/bin  
# ./startsunvts -h
```

Or,

```
# cd /usr/sunvts/bin  
# ./startsunvts --help
```

Interpreting SunVTS Messages

There are many messages for each of the many tests that run under SunVTS. For this reason, it is difficult to document the implication of each message that you might see. Most messages contain all the explanatory text that is possible for each event that is encountered. Some messages are information messages (INFO, VERBOSE, WARNING) which do not represent an error, while some messages are error messages (ERROR and FATAL), indicating that the test detected a failure. This section explains generic test message information.

TABLE 2-3 SunVTS Information Message

```
12/16/07 23:02:52 ctech114 SunVTS7.0build80: VTSID 0
Processor.fputest.INFO All: Version: fputest 7.0 2007/12/04 (7.0
build80)

12/16/07 23:03:13 ctech114 SunVTS7.0build80: VTSID 0
Processor.iutest.INFO cpu-unit: Version: iutest 7.0 2007/12/04 (7.0
build80)

12/16/07 23:03:51 ctech114 SunVTS7.0build80: VTSID 0
Processor.iutest.INFO cpu-unit: Version: iutest 7.0 2007/12/04 (7.0
build80)

12/16/07 23:04:04 ctech114 SunVTS7.0build80: VTSID 0 cddvdtest.INFO
rdsk/c0t0d0s0: Version: cddvdtest 7.0 2007/12/04 (7.0 build80)

12/16/07 23:04:04 ctech114 SunVTS7.0build80: VTSID 2102 cddvdtest.INFO
rdsk/c0t0d0s0: "Media Type = DVD-ROM, Block Size = 2048,
Capacity(blocks) = 1128150,
Max on disk= 4284160
"

12/16/07 23:04:12 ctech114 SunVTS7.0build80: VTSID 0 vmemtest.INFO
kmem:
Version: vmemtest 7.0 2007/12/04 (7.0 build80)

12/16/07 23:04:12 ctech114 SunVTS7.0build80: VTSID 0 pmemtest.INFO
mem:
Version: pmemtest 7.0 2007/12/04 (7.0 build80)

12/16/07 23:04:27 ctech114 SunVTS7.0build80: VTSID 0
Processor.iutest.INFO cpu-unit: Version: iutest 7.0 2007/12/04 (7.0
build80)
```

TABLE 2-4 SunVTS Error Message

```
10/08/07 23:47:31 ctech114 SunVTS7.0build75: VTSID 8003
Interconnect.nettest.FATAL ce0(/pci@1c,600000/network@2): "System
call
recvfrom returned -1 (Connect
ion refused)."
```

```
10/08/07 23:56:40 ctech114 SunVTS7.0build75: VTSID 6049
Graphics.m64test.FATAL fbs/m640: Can't open display
  Probable_Cause(s): <Permission denial from Host> <System error> <X
Server is not running>Recommended_Actions(s): <run xhost(1)> <If X
server is not running
, run X server> <If problem persists, call your authorized SUN service
provider>
```

```
10/09/07 03:43:58 ctech114 SunVTS7.0build75: VTSID 8003
Network.nettest.FATAL ce0(/pci@1c,600000/network@2): "System call
recvfrom returned -1 (Connection r
efused)."
```

```
10/09/07 10:50:00 ctech114 SunVTS7.0build75: VTSID 8058
Ioports.serialtest.FATAL su0: "Failed ioctl TIOCSLOOP, device =
/dev/term/a. Probable Cause: Port i
s in use. Please verify if a device/cable is connected to the port.
"
```

```
10/09/07 10:55:09 ctech114 SunVTS7.0build75: VTSID 8003
Network.nettest.FATAL ce0(/pci@1c,600000/network@2): "System call
recvfrom returned -1 (Connection r
efused)."
```

```
10/09/07 10:59:28 ctech114 SunVTS7.0build75: VTSID 6049
Graphics.m64test.FATAL fbs/m640: Can't open display
  Probable_Cause(s): <Permission denial from Host> <System error> <X
Server is not running>Recommended_Actions(s): <run xhost(1)> <If X
server is not running
, run X server> <If problem persists, call your authorized SUN service
provider>
```

```
10/09/07 19:07:20 ctech114 SunVTS7.0build75: VTSID 8003
Interconnect.nettest.FATAL ce0(/pci@1c,600000/network@2): "System
call
recvfrom returned -1 (Connect
ion refused)."
```

Whenever possible, the error message supplies a message type, followed by the message text, one or more probable causes, and one or more recommended corrective actions. [TABLE 2-5](#) describes the types of messages you might encounter. All messages are displayed in the Message panel, and most are logged in one of the SunVTS logs (Info or Error).

TABLE 2-5 SunVTS Message Types

Message Type	Log File	Description
INFO	Info log	Displayed when a non-error test event occurs.
ERROR	Error and Info logs	Displayed when a test detects an error, often an indication of a problem with a specific function or feature of the device under test.
FATAL	Error and Info logs	Displayed when a test encounters a severe condition that might cause the test to exit, such as a failure to open a device. These messages generally do not indicate a hardware failure.
VERBOSE	Verbose Log	Test progression messages that are displayed when the verbose feature is enabled.
WARNING	Info log	These messages warn you of a condition that might prevent the physical test to run with the test's full capabilities. These warning messages do not necessarily indicate an error.

SunVTS on LDoms Enabled Systems

SunVTS 7.0 functionality is available in the control domain and guest domains on LDoms-enabled Sun SPARC servers.

On the I/O tests side, `Disk` and `Network` will show up. I/O ports will show up if a virtual keyboard is present.

If the `cryptotest` fails, check to see if your guest domain has MAUs configured. To do this, type the following:

```
# ./ldm list-bindings guest-domain-name | grep -i mau
```

This command displays output if `mau` is configured.

Backward Compatibility

The SUNWvts (harness and APIs) package guarantees backward compatibility across minor update releases. SunVTS software can be broken across major releases, though.

Displaying Devices Using the VTSpoke Utility

The `vtspoke` command lists all of the system testable devices, associated configuration information, and corresponding hardware tests.

```
% /usr/sunvts/bin/vtspoke
cmttest(cmt)
CMTstress
Number of testable CMT:8

cpustest(CPU)
CPU Test

cryptotest(ncp0)
Crypto Accel Bulk 1.0
Model: N2CP Crypto Accelerator
Device Kind :N2CP-Crypto-Device

disktest(c0t0d0)
Disk-c0t0d0
Capacity: 68.35GB
Controller: mpt0
Vendor: FUJITSU
SUN Id: MAY2073RCSUN72G
Firmware Rev: 0501
Serial Number: 0709S08SHK
Device Kind :disk

cddvdtest(c1t0d0)
CD-ROM(c1t0d0)
Controller: uata0

dtlbtest(dtlb)
Data TLB Test
```



```

fputest(FPU)
Floating Point Unit Test

hsclbtest(hsclb)
Host to System Controller Loopback Test
Host to System Controller Loopback Test

iobustest(iobus)
IOBUS
platform = SUNW Sun-Fire-T200

iutest(IU)
Integer Unit Test

l1dcachetest(l1cache)
Level 1 Cache Test

l2sramtest(l2sram)
External Cache Test

mpptest(mp)
MPstress
Number of processors:32

netlbtest(e1000g1)
lgbaseT
Port Address: Unknown
Host ID: 8481fe02

netlbtest(e1000g2)
lgbaseT
Port Address: Unknown
Host ID: 8481fe02

netlbtest(e1000g3)
lgbaseT
Port Address: Unknown
Host ID: 8481fe02

nettest(e1000g0)
1kbaseT
Host_Name:
Host Address: 10.12.150.50
Host ID: 8481fe02
Domain Name: blr03-01.India.Sun.COM

pmemtest(mem)
PhysicalMemory

```

```
System memory size:32760MB

ramtest(mem)
Memory
Total Physical Memory : 32760MB

serialtest(su0)
SerialPorts
Port a = su0 = /dev/term/a = /devices/...su...a

systest(system)
System Stress Test
System Configuration= Sun Microsystems sun4v
Memory size= 32760 Megabytes
System clock-frequency= 200 MHz

usbtest(kbd)
USB Device
Device information: USB Keyboard

vmemtest(kmem)
VirtualMemory
Total Swap: 27677MB
```

Using the SunVTS Browser User Interface

This chapter describes using the SunVTS 7.0 browser user interfaces (BUI). Topics include:

- [“Starting the Browser User Interface” on page 25](#)
- [“Host Machine View” on page 28](#)
- [“Test Group Screen” on page 35](#)
- [“Logs Screen” on page 52](#)

Starting the Browser User Interface

Note – SunVTS must first be started before selecting a user interface. See [“Starting SunVTS” on page 14](#). The BUI may not be supported in future releases of SunVTS.

The SunVTS 7.0 browser user interface (BUI) is the Java Web Console. The Java Web Console provides a common point for Sun web-based system management applications to be registered and accessed. A common platform login is provided as well as a Start page to access all registered applications under these categories: Systems, Storage, Services, Desktop Applications, and Other.

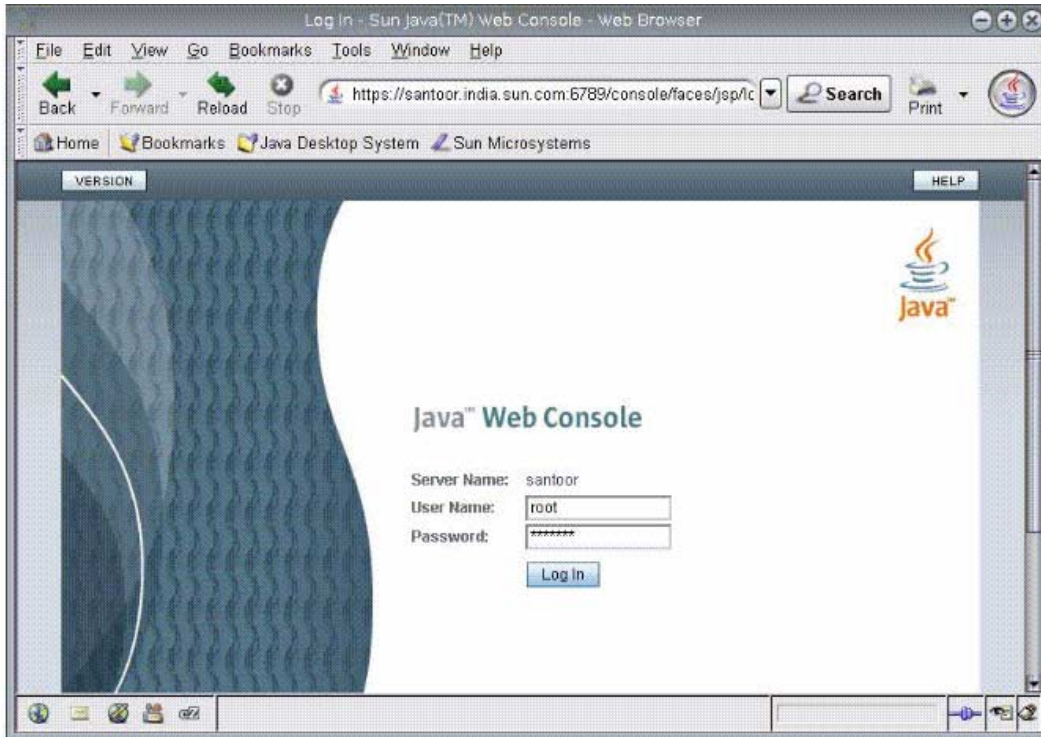
Using the browser, start the URL where the Java Web Console is installed as `https://hostname:6789`. For example,
`https://santoor.india.sun.com:6789`.

▼ To Start the SunVTS Browser User Interface

1. **Using the browser, start the URL where the Java Web Console is installed as**
`https://hostname:6789.`

The default login window (FIGURE 3-1) is displayed.

FIGURE 3-1 Sun Java Web Login Window



2. **Type the user name.**
3. **Type the password.**
4. **Click Log in.**
The Java Web Console is displayed (FIGURE 3-2).
5. **Click the SunVTS application listed under Systems.**
The Host Machine View window appears (FIGURE 3-3).

FIGURE 3-2 Sun Java Web Console

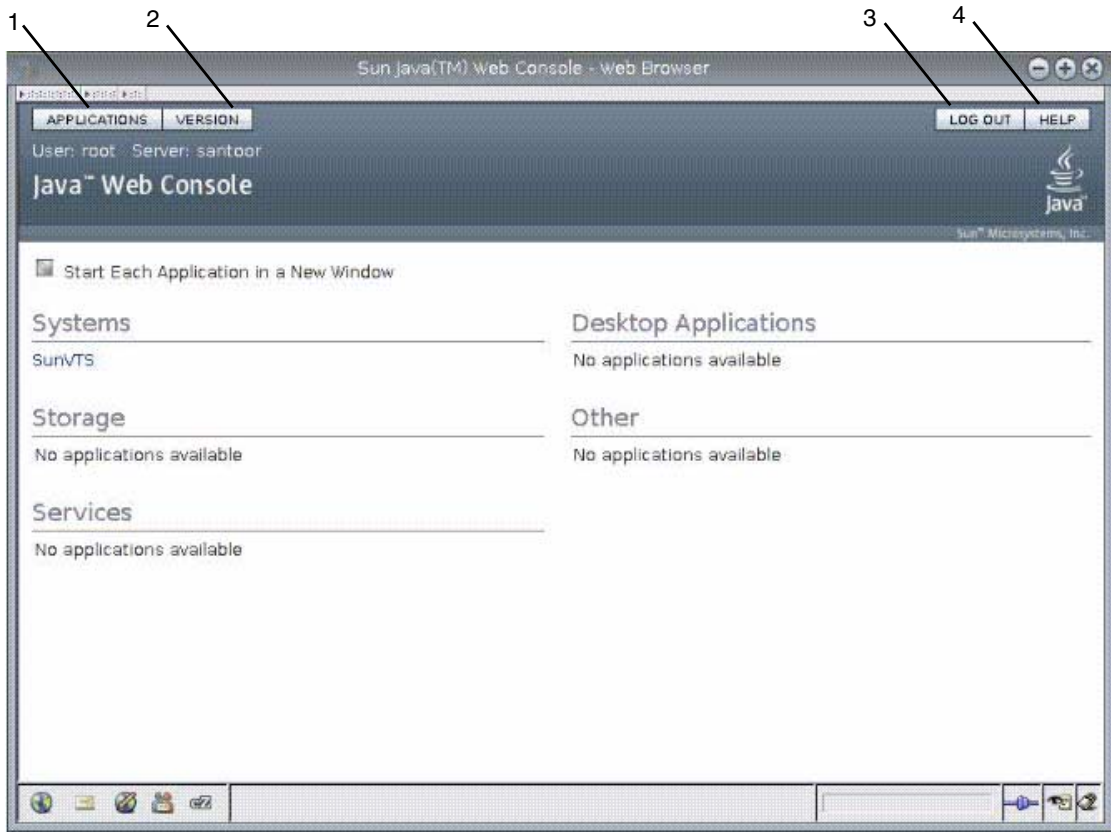


TABLE 3-1 describes the Java Web Console window buttons.

TABLE 3-1 Java Web Console Button

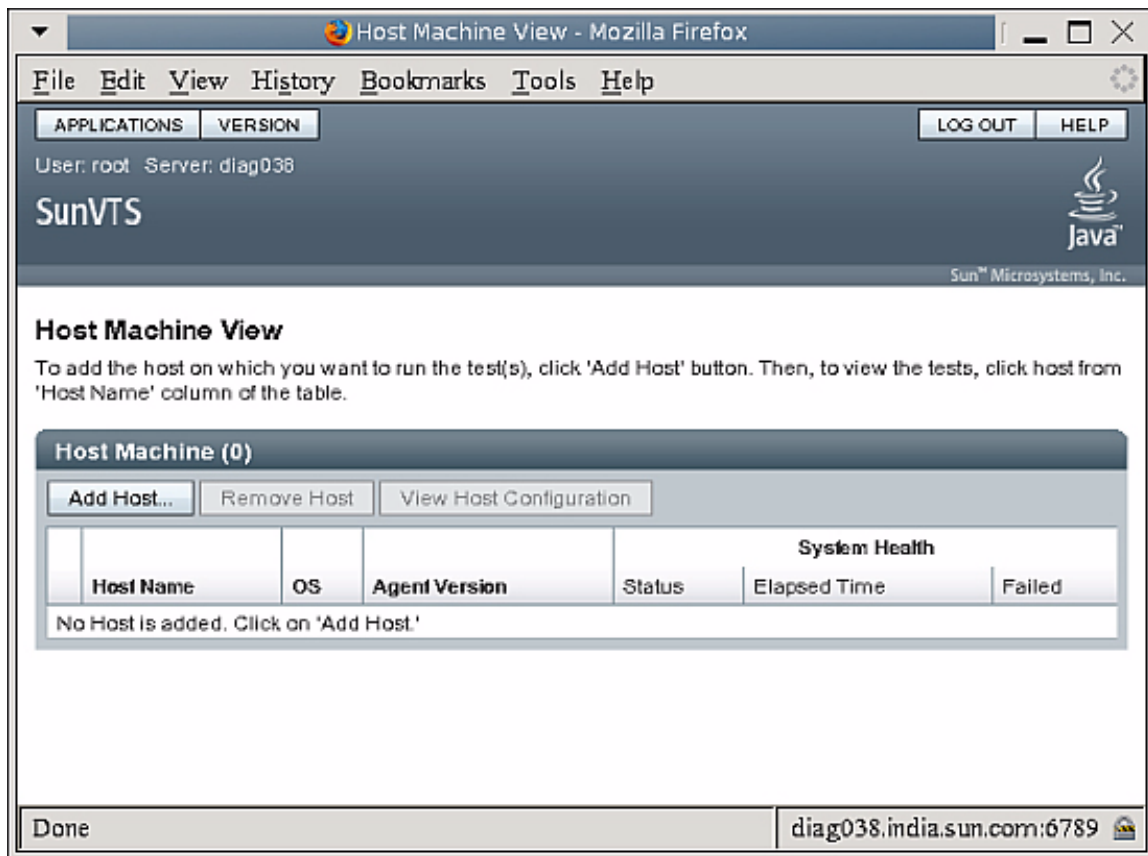
Item	Button	Description
1	Applications	Displays all the applications registered with Sun Java Web Console (FIGURE 3-2). Also displays the user that is currently logged in and the middle server name where the user has logged in.
2	Version	Opens the SunVTS version window.
3	Logout	Logs out of the SunVTS console.
4	Help	Opens the Help window.

Host Machine View

The Host Machine View screen (FIGURE 3-3) facilitates adding, removing, and viewing the configuration of a Host to be diagnosed.

The host to be configured should have both the middle server and `vtsk` installed. Ensure that the `vtS` kernel is running on the hosts that you want to configure for system testing. Before initiating any tests or monitoring the test status, you must first add the host.

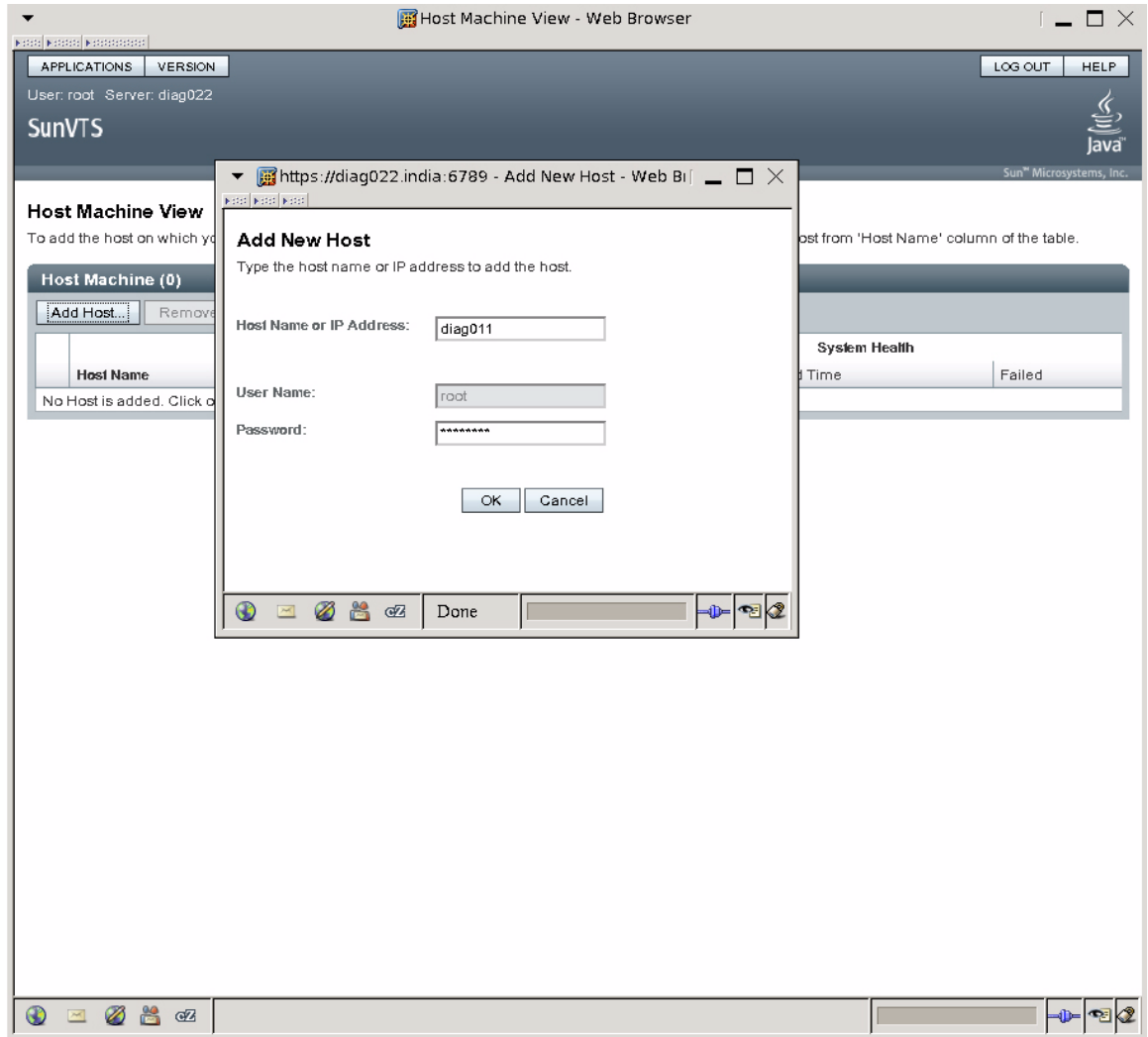
FIGURE 3-3 Host Machine View Window



▼ To Add a Host

1. From the Host Machine View, click Add Host (FIGURE 3-3).
The Add New Host dialog box is displayed (FIGURE 3-4).

FIGURE 3-4 Host Machine View – Add New Host Dialog Box

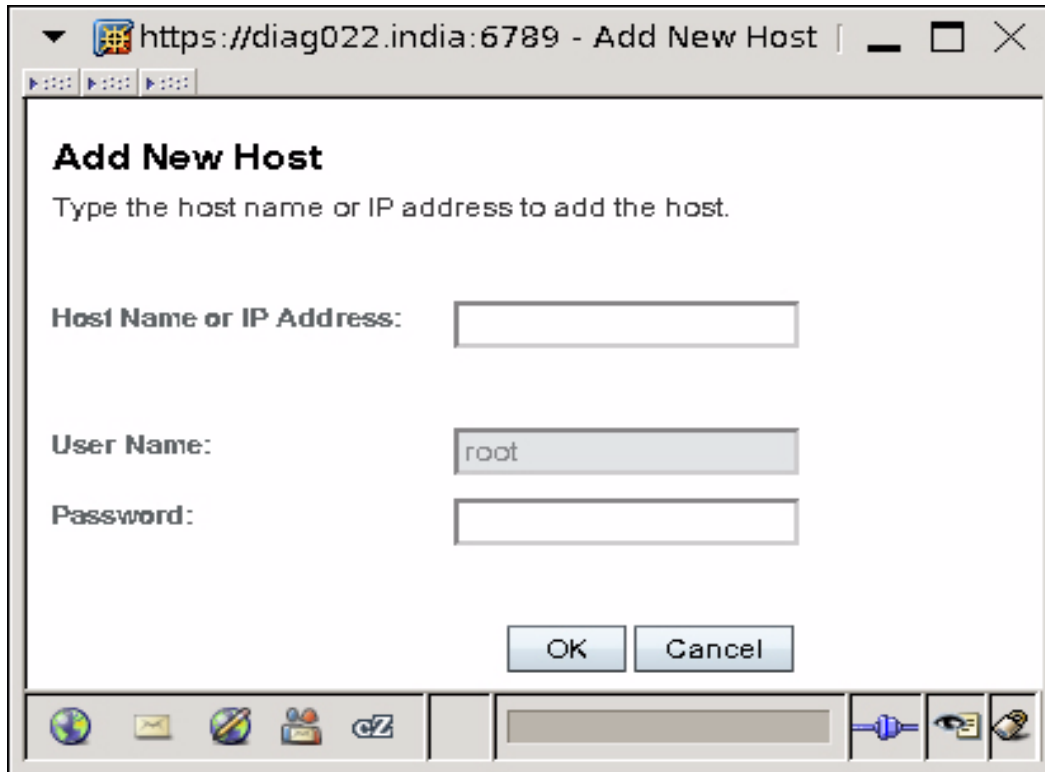


2. Type the IP address or host name.
3. Log in as root and type the user name and password.

4. Click OK.

The Add New Host screen is displayed (FIGURE 3-5).

FIGURE 3-5 Add New Host Window



If vtshk and JAVA bridge is not running on the host or agent machine, or in case of some other error, an alert message of Login Failed is displayed on the Add New Host pop-up (possibly from a login failure).

Note – You can add only one host at a time using the Add Host pop-up window. That host is the system under test.

When you add a host, an entry appears on the Host Machine View table (FIGURE 3-6).

FIGURE 3-6 Host Machine View Window

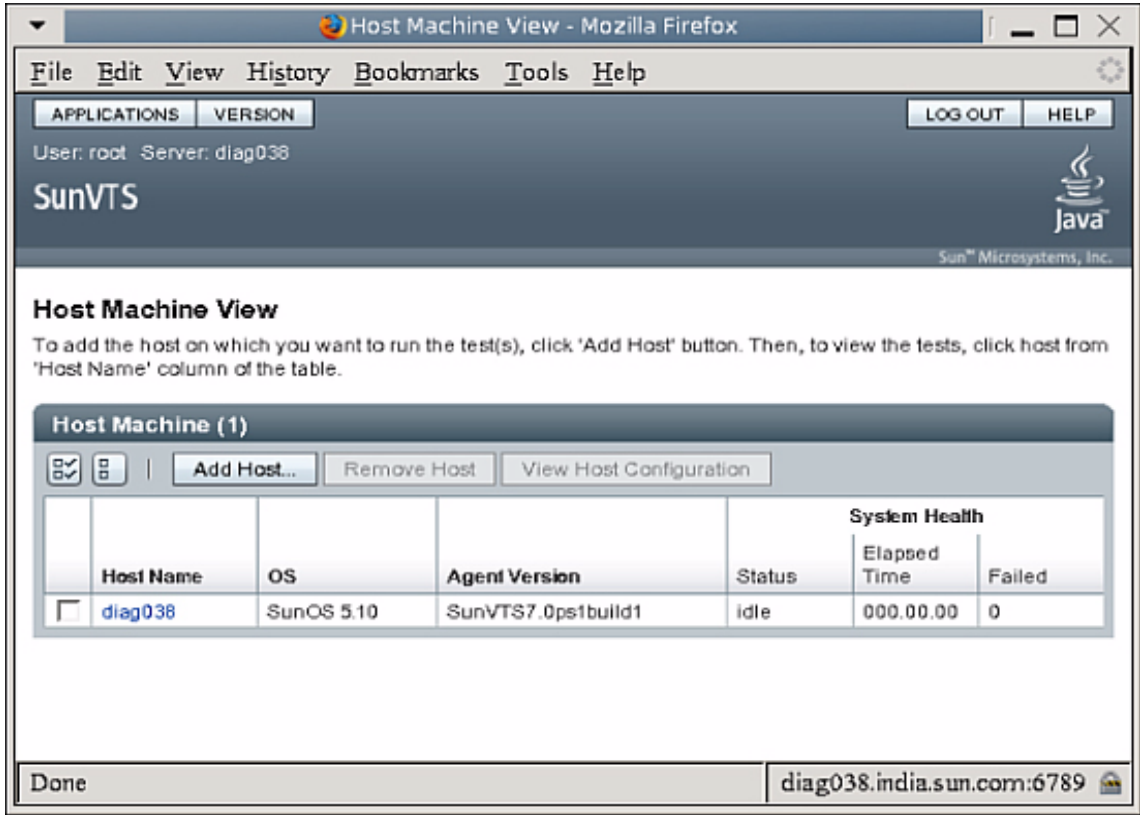


TABLE 3-2 shows the Host Machine View screen descriptions.

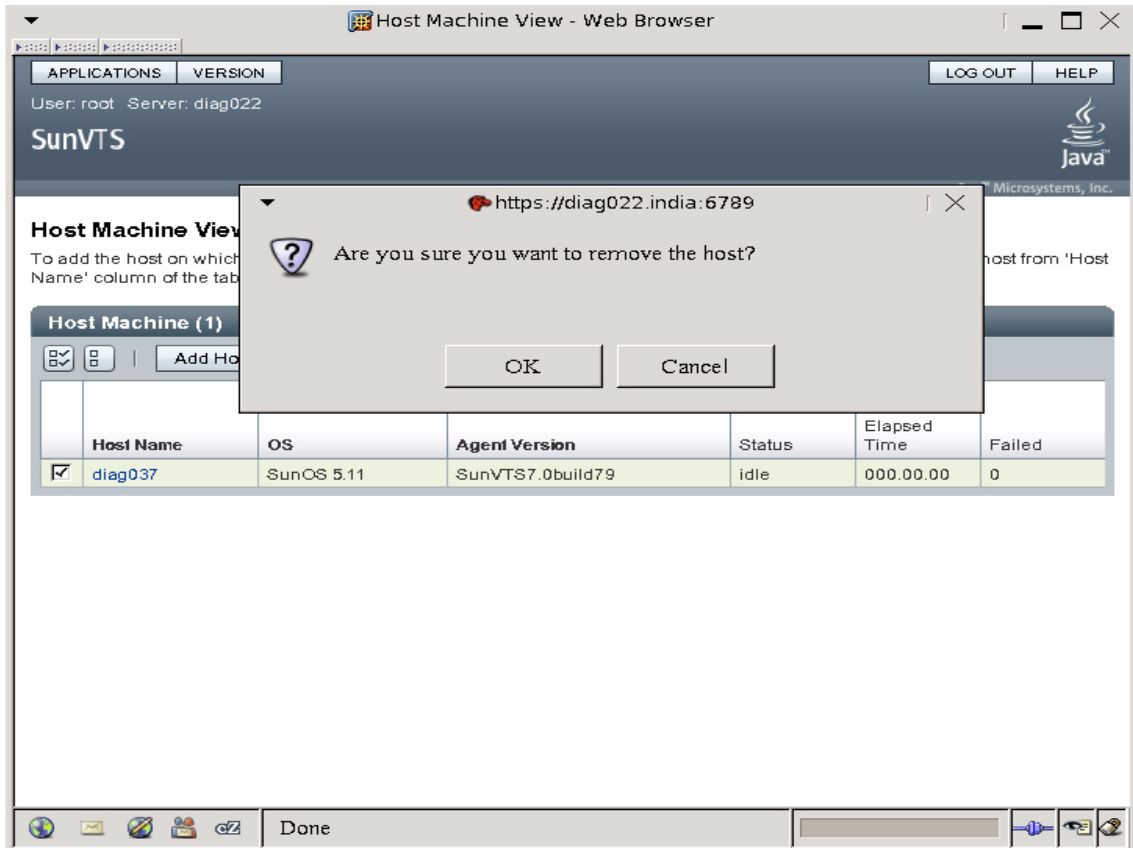
TABLE 3-2 Host Machine View Screen

Column	Description
Host Name	Displays the currently logged host machine.
OS	Displays the operating system installed in the host machine.
Agent version	Displays the SunVTS version installed in the host machine.
Status	Shows the global system status (whether testing, stopping, or idle) with subcolumns for status, elapsed time, and total number of errors incurred at any point of testing.

▼ To Remove a Host

1. Click one or more boxes next to the host you want to remove.
The Remove Host button is highlighted.
2. Click the Remove Host button, and click OK on the confirmation window (FIGURE 3-7).

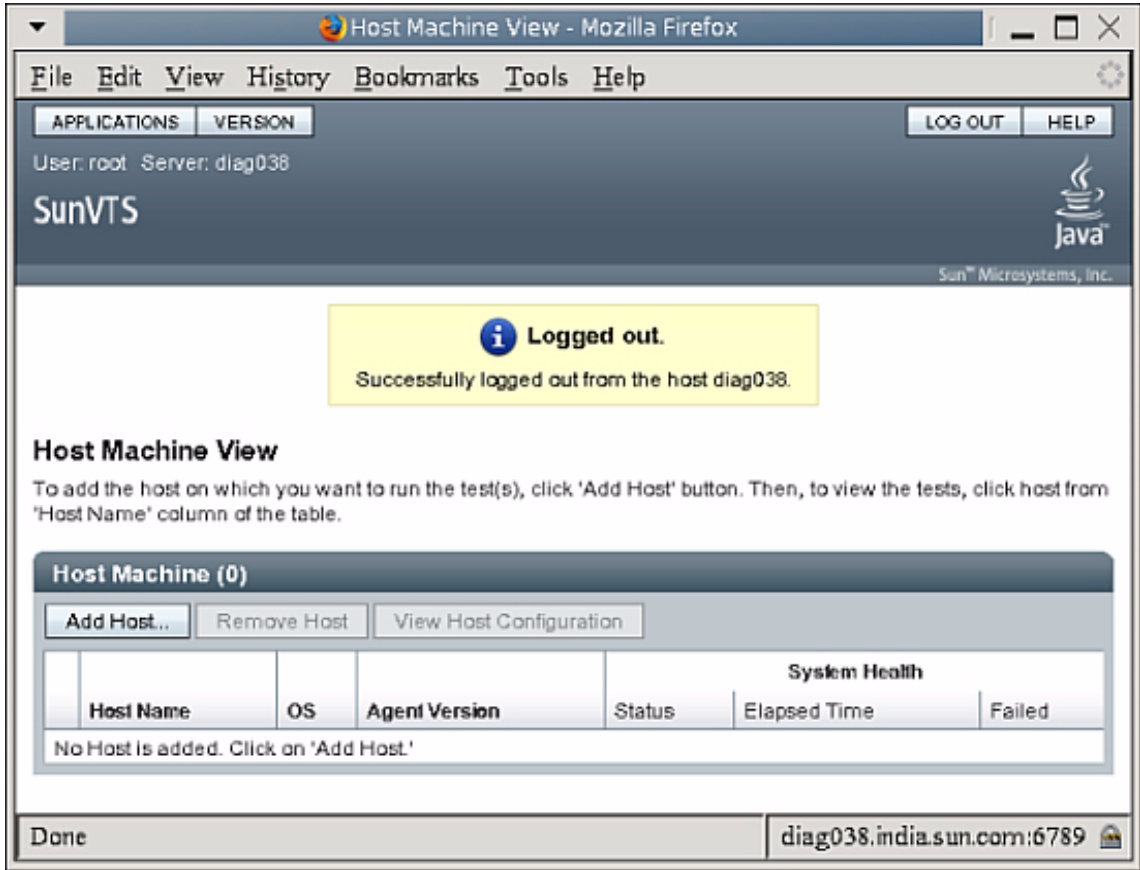
FIGURE 3-7 Remove Host Confirm Window



3. Click OK to remove the host.

The host is removed and an alert message is displayed that indicates a successful logout (FIGURE 3-8).

FIGURE 3-8 Successful Log Out Window



▼ To View Host Configuration

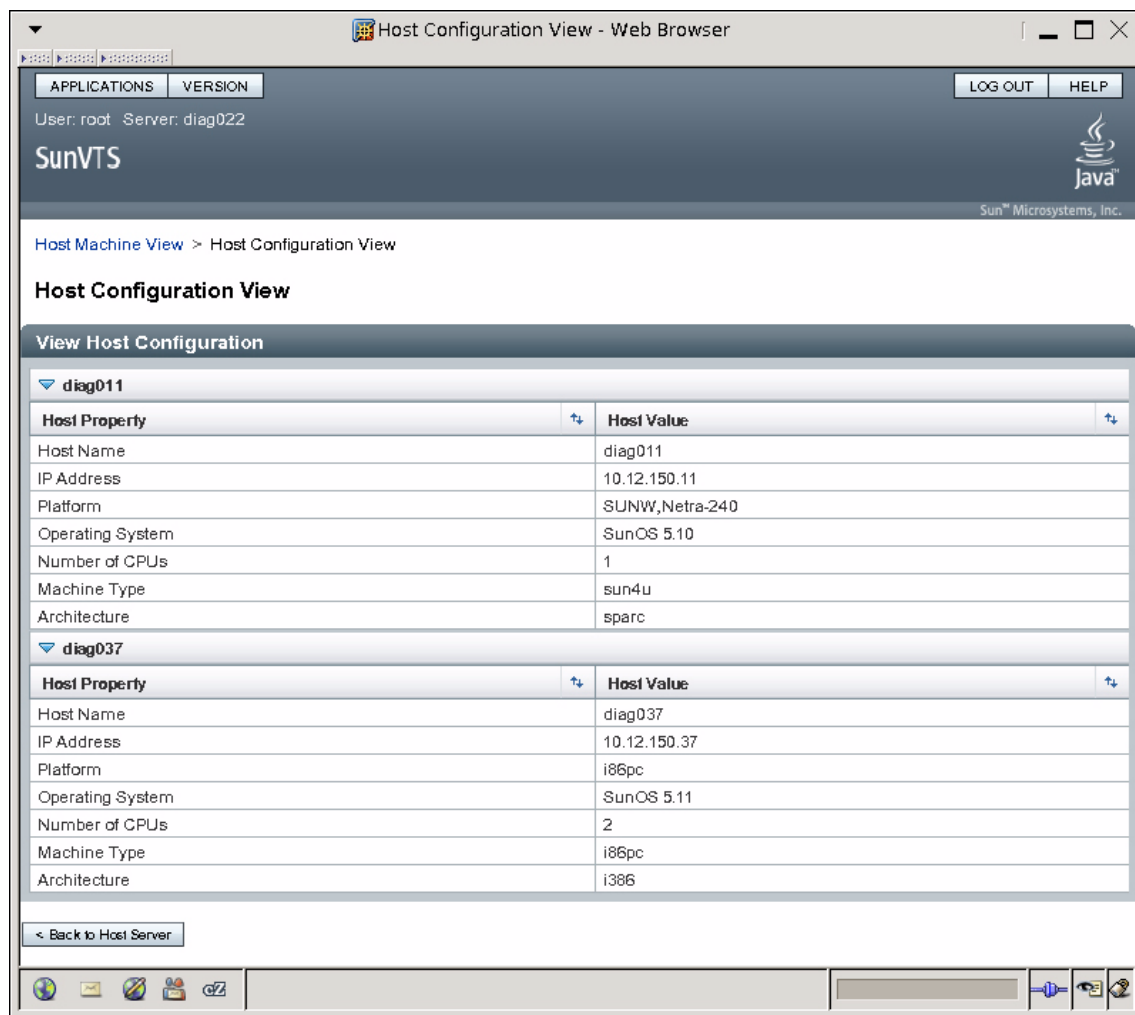
1. Click one or more boxes next to the hosts whose configuration details you want to view.

The View Host Configuration button is highlighted.

2. Click the View Host configuration button.

The details displayed are host name, IP address, platform, operating system, number of CPUs, machine type, and architecture of the selected system (FIGURE 3-9).

FIGURE 3-9 Host Configuration View Window



3. Click the Back to Host Server button to return to the Host Machine View.

An alternate way to go back to the parent page is by using the navigation link at the top of the window.

Test Group Screen

The Test Group window ([FIGURE 3-10](#)) lists the tests that are registered with the host machine.

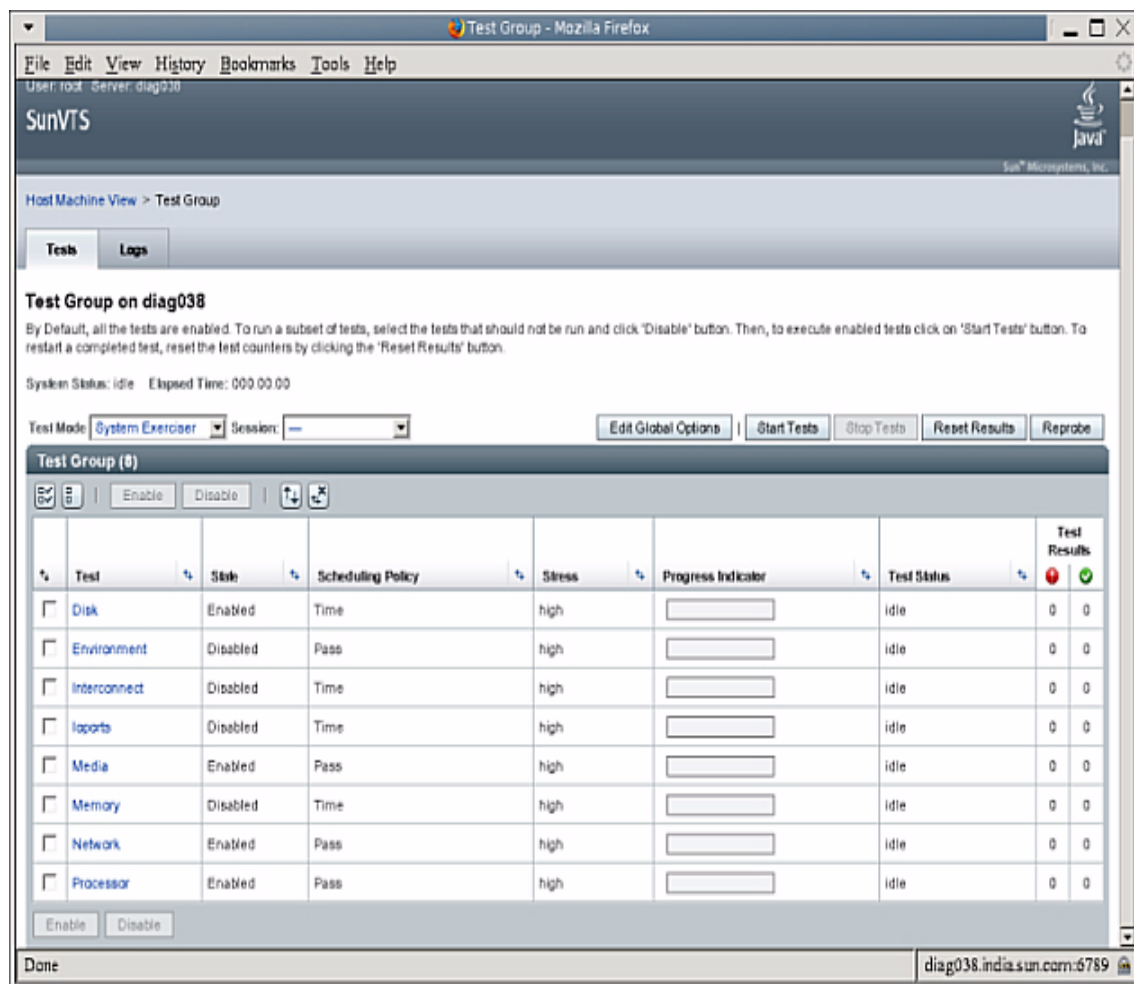
1. Click the Hostname link in the Host Machine View.

The Test Group window is displayed. The Tests tab is associated with the Test Group View.

The host machine name can be viewed on the heading of the window.

2. Click on the Host machine View link to navigate back to the parent window.

FIGURE 3-10 Test Group Window



Test Group

The Test Group table (FIGURE 3-10) displays the tests registered with the host. This window also has the following features that allow you to:

- Set the testing modes, schedule the tests, and monitor the testing status of each of the tests.
- Store any selection settings in the Session Utility. Later, you can load the stored settings to perform the system testing.

- Select the tests to run for testing. Each row in the table has a test with the default set of test specific values and a progress bar (only for the selected tests once the testing is started).
- View the progress bar to see the percentage of the test completed.

FIGURE 3-11 shows a Test Group Screen under test.

FIGURE 3-11 Test Group Window Under Test

Test Group on diag038

By Default, all the tests are enabled. To run a subset of tests, select the tests that should not be run and click 'Disable' button. Then, to execute enabled tests click on 'Start Tests' button. To restart a completed test, reset the test counters by clicking the 'Reset Results' button.

System Status: testing Elapsed Time: 000 01:41

Test Mode: System Exerciser Session: —

Edit Global Options | Start Tests | Stop Tests | Reset Results | Reprobe

Test Group (8)

%	Test	%	State	%	Scheduling Policy	%	Stress	%	Progress Indicator	%	Test Status	%	Test Results
<input type="checkbox"/>	*Disk		Enabled		Time		high		<div style="width: 50%;"></div>		testing		0 0
<input type="checkbox"/>	Environment		Disabled		Pass		high		<div style="width: 0%;"></div>		idle		0 0
<input type="checkbox"/>	Interconnect		Disabled		Time		high		<div style="width: 0%;"></div>		idle		0 0
<input type="checkbox"/>	Ioparts		Disabled		Time		high		<div style="width: 0%;"></div>		idle		0 0
<input type="checkbox"/>	Media		Enabled		Pass		high		<div style="width: 0%;"></div>		failed		1 0
<input type="checkbox"/>	Memory		Disabled		Time		high		<div style="width: 0%;"></div>		idle		0 0
<input type="checkbox"/>	*Network		Enabled		Pass		high		<div style="width: 50%;"></div>		testing		0 1
<input type="checkbox"/>	*Processor		Enabled		Pass		high		<div style="width: 0%;"></div>		testing		0 0

Enable | Disable

Done

diag038.india.sun.com:5789

Test Table Contents

This section describes the parameters available in setting up a test. Select parameters from the Test Group screen (FIGURE 3-10) and Test Options screen (FIGURE 3-14).

The Test Group screen (FIGURE 3-10) displays the name of the system being viewed for status (*santoor* in this example). The following items describe the table:

■ Test

Tests can be any of the following: Memory, Disk, Network, HBA, Graphics, Processor, IOports, interconnect, and Media. Click on the test link to set the test options; stress level, scheduling policy (time-based or in terms of test passes), and error limit.

■ State

This column shows the state of the test at a given time (enabled or disabled).

■ Scheduling Policy

On the Test Options screen (FIGURE 3-14), select Test Time or Test Passes based on the testing requirement. In the case of Test Time, specify the time in minutes in the text box (range 0 to 99999). For Test Pass, specify the number of passes in the text box (range 0 to 99999). By default, Test Time is the scheduling policy and the default value for both (Time or Passes) is zero.

Select Test Passes when time is not a constraint and is being used to reproduce an error on the specific test.

■ Stress

Select low or high (FIGURE 3-14).

■ Progress

The Progress bar in the Test Group table (FIGURE 3-10) indicates the percentage of the test run completed. A tool tip over the progress bar shows the progress of the test in percentage and the current state of the test.

■ Test Status

The Test Status column in the Test Group table (FIGURE 3-10) shows the status of the test, either idle, testing, analyzing, not supported, stopping, failed, or completed, along with the passes and error count. Each status item is described as follows.

Idle – A test can be idle for any of the following reasons:

- Initial status of a test before testing.
- SunVTS resets the tests.
- You click the Stop button.
- The global time limit completes and all the tests stop.

Not supported – For the given stress level, a test might not be able to run for any of the following reasons:

- The rules file does not define any task.
- The tests defined under the task (in the rules file) were not able to probe any of the devices.

Analyzing – A test is running after completion of each pass that checks for certain terminating parameters such as max time, max pass, and max error to analyze if it can run for the next pass. The Analyzing status lasts for 2 to 3 seconds.

Testing – The test is currently executing.

Completed – The test has completed its execution. The completion factor is based on the limits set from the scheduling policy, (time or pass limit).

Failed – The test stopped because the test errors reached the maximum error limit.

Waiting for swap – If there is not enough swap available, you can stop a test and schedule the test execution for later.

Stopping – When you press the Stop button, the test status shows the transition from stopping status to idle status.

■ Test Results

The Test Results column displays test results in two subcolumns. The first subcolumn displays the number of test errors and second displays the number of test successes.

Test Group Button Functions

You can perform various functions on the Test Group table ([FIGURE 3-10](#)) using the buttons Enable, Disable, Start Tests, Stop Tests, Edit Global Options, Reset Results and Reprobe. [TABLE 3-3](#) describes the button functions.

TABLE 3-3 Test Group Screen

Button	Description
Enable	All tests are enabled. By default, all tests are enabled. To verify which tests are enabled and which are not, look for the state of the tests under the State column corresponding to the test.
Disable	All tests are disabled. To verify if a particular test is disabled, look for the state of the test under the State column corresponding to the test.
Start Tests	Starts all the tests that are enabled. By default, all the tests are enabled. To run a particular test, disable other tests by clicking the corresponding checkbox; then, click the Disable button. To start a particular test when other tests are already running, simply select the checkbox for the test and click Enable.

TABLE 3-3 Test Group Screen (*Continued*)

Button	Description
Stop Tests	Stops all the tests that are started. To stop a particular test that is running, select the checkbox for the test and click Disable.
Edit Global Options	Sets options that you can apply to all the tests globally. These options can be used to schedule the total time of the testing session.
Reset Results	Resets the status counters (pass, elapsed time, and error count of a test is reset to zero) and progress bar to an idle state. Before starting any new testing session. Click Reset.
Reprobe	Reprobes the system for devices currently available.

Test Modes

Select the Test Mode depending upon the testing requirement. Test Mode is located at the top left corner of the Test Group table. You can select one of the three test modes (Online Stress, System Exerciser, and Component Stress) from the drop-down menu.

See [“Test Modes” on page 4 in Chapter 1.](#)

By default the Test Mode is set to System Exerciser mode.

Test Session

The Session feature is very useful when you want to automate SunVTS 7.0 with some defined options. Instead of setting up the combination of test, global, or mode options each time you need to diagnose the host machine, you can create a session file with the combination of options. Then, save your test preferences with a session name. The session is saved in XML format. Load the session file and the saved option settings are used.

▼ To Save a Session

For selected test modes, you can save chosen tests with configured option values. This feature enables you to repeat tests in the same configuration for the same machine or a different machine.

To save a test session:

- 1. Configure SunVTS for the test session that you want to save.**

2. From the Test Group window, select a Save Session option from the Session menu.

The Save Session window is displayed ([FIGURE 3-12](#)).

3. Select or type a name in the Session Name text field.

You can also choose an existing session name from the list.

Note – Do not include a space in the session name input.

4. Select a Test Mode as the default active Test Mode for the Session.

5. Select the Overwrite Session option to overwrite the existing session with the new session.

6. Select the type of session: Generic or Host Specific.

7. Click OK to save the session.

The Save Session window is closed, and your session configurations are saved for future use.

FIGURE 3-12 Save Session Screen

Sessions on diag060
List of currently saved sessions

Session:

Session Name:

Test Mode:

Session Type:

Overwrite Session ☐

Done

▼ To List a Session

You can view all the sessions existing for the machine. To list sessions:

1. **From the test group window, select the List Sessions option from the Session menu.**

The List Sessions window is displayed with the list of all the existing sessions

2. **Click Close to close the List Session Window.**

▼ To Load a Session

This option allows you to choose desired session. The test configuration and option values for tests execution is set based on selected session. To load sessions:

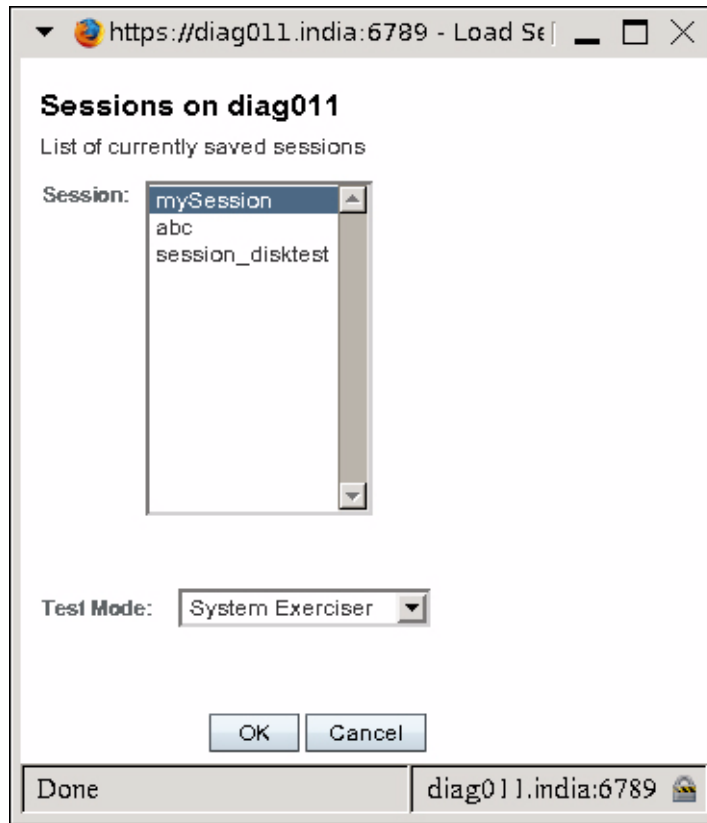
1. **From the Test Group window, select a Load Session option from the Session menu.**

The Load Session window is displayed.

2. **Select an existing session name to load.**
3. **Select a Test Mode as the default Test Mode to load the session.**
4. **Click OK to Load the session.**

The Load Session window is closed, and the test session configuration is loaded into SunVTS. SunVTS will update and refresh the values on the BUI. The BUI will show the name of the loaded session. You can use this configuration, or modify it, before you start the testing ([FIGURE 3-13](#)).

FIGURE 3-13 Load Session Screen



▼ To Delete a Session

If you no longer need a specific test configuration, you can delete the session associated with it. To delete a session:

1. **From the Test Group window, select Delete Sessions from Session drop-down menu.**

The Delete Session window is displayed with the list of all existing sessions.

2. **Select the Session that you want to delete from the List of Sessions.**
3. **Click OK to delete the selected Session.**

The Delete Session window is closed, and the selected session is deleted.

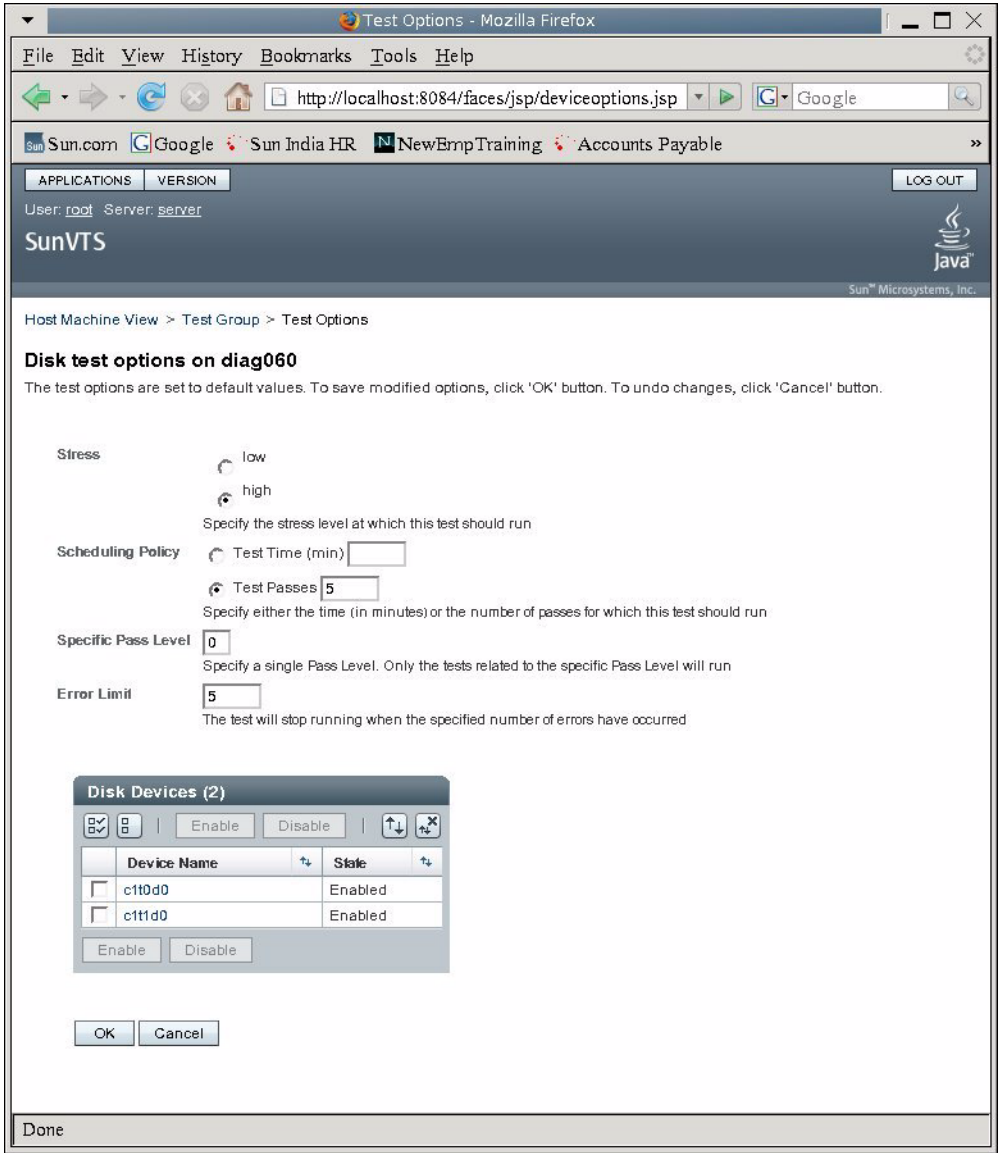
Reset Session

If you loaded or saved a session but then you decide to run the tests with the default test configuration, you can use this option. This clears the session values and settings and loads the default parameters.

Test Options Screen

The Test Options screen ([FIGURE 3-14](#)) appears when you click the Test Name link from the Test Group ([FIGURE 3-10](#)) table in the Test column.

FIGURE 3-14 Test Options Screen



After selecting a test from the Test Groups table, select the test options that you need. For example: select Disk Test Options (FIGURE 3-15), or Interconnect Test Options (FIGURE 3-16).

FIGURE 3-15 Disk Test Options Screen

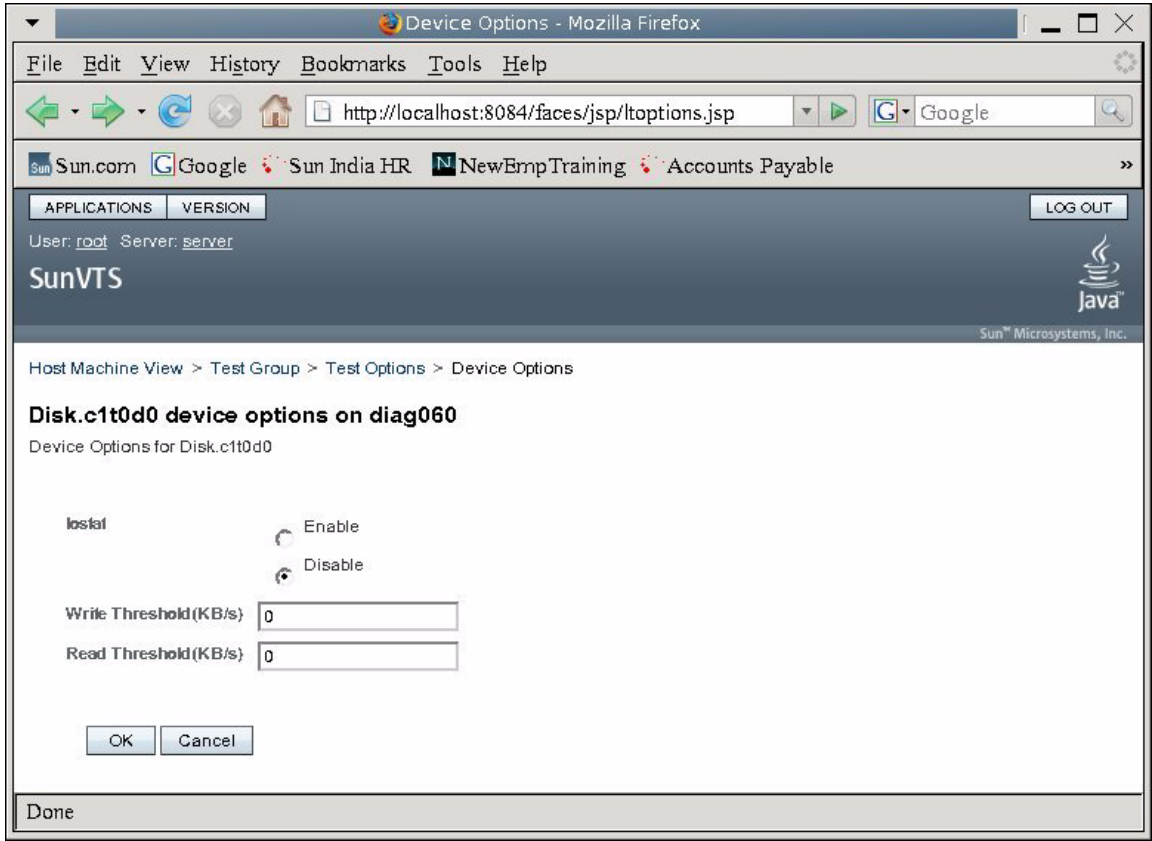
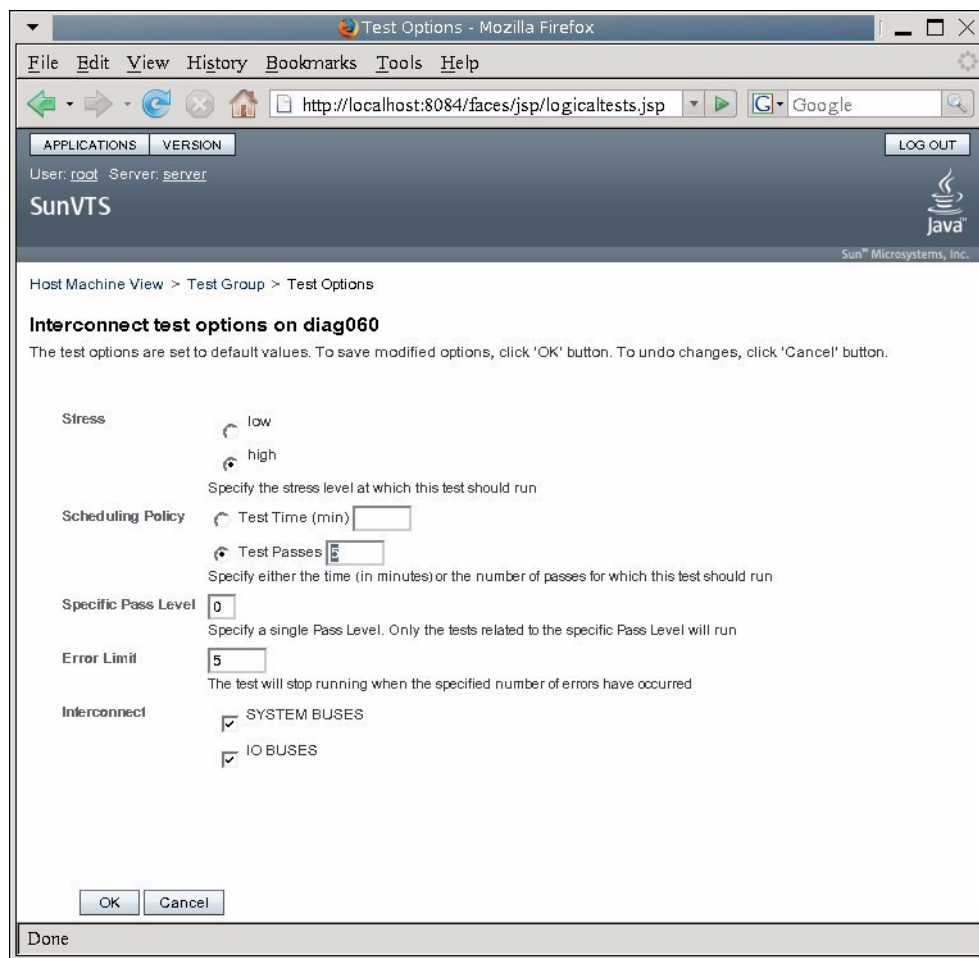


FIGURE 3-16 Interconnect Test Options Screen



▼ To Change Test Options

1. Specify the stress level - low or high, depending upon your requirement.

The default stress values are based on the system configuration. If the system has a small memory configuration, the default stress is set to “low” level. Otherwise, the default stress is “high” level. SunVTS considers a system as a small memory configuration if it has less than or equal to 2GB of memory (for x86), and less than or equal to 4GB (for SPARC).

2. Choose the Scheduling Policy between Test Time or Test Passes.

Depending on the selection, provide the time in minutes or number of passes.

- Online Stress Default - Scheduling Policy is Test Pass, and the default Test Pass is 1. If the scheduling policy is changed to Test Time, the default Test Time will be 240 minutes.
- System Exerciser Default - the Scheduling Policy is Test Time, and the default Test Time is 0 (or infinite). If the scheduling policy is changed to Test Pass, the default Test Pass is 5.
- Component Stress Default - Scheduling Policy is Test Pass, and the default Test Pass is 5. If the scheduling policy is changed to Test Time, the default Test Time will be 240 minutes.

3. Choose a Specific Pass Level.

You can choose a specific pass level and only the test related to that specific pass level will run. The valid range of pass levels are listed in the enclosed bracket next to the specific pass level tag. By default, the value of Specific Pass level is 0 and means that no specific pass level has been selected.

4. Choose an Error Limit.

Specify the number of errors beyond which the test should stop executing. By default, the Error Limit is set to 1. An error limit of 0 would mean that the test continues to run with no limits to error count.

5. Choose specific test device options (optional).

Some Tests export specific options that allow you to select the test devices and modify the device options.

a. **Select the check box corresponding to the device and click the “enable” or “disable” button to select/de-select the device. By default, all the devices will be enabled or disabled.**

b. **To change the device options, click on the device link.**

The test option selection window is displayed.

c. **Edit the test options.**

6. Click OK to confirm the changes and to return to the parent page.

The Cancel button cancels the changes and returns to the parent page.

Note – Setting a Test Time to 0 (zero) or Test Pass to 0 (zero) indicates that the test must run for an infinite time or infinite pass respectively. In such a situation the progress bar will not show significant change. You must specify the Test Time or Test Pass with an integer value to see correct indication of the progress bar.

▼ To Set Global Options

There are certain options that can be applied to all the tests globally. These options can be used to schedule the total time of the testing session. To edit the Global options:

1. Click on the **Edit Global Options** button on the **Test Group** screen.

The Global Test Options screen is displayed (FIGURE 3-17).

FIGURE 3-17 Global Test Options Screen

The screenshot shows the 'Global Test Options for the host ctech59a' window. At the top, there's a header with 'APPLICATIONS' and 'VERSION' tabs, and 'LOG OUT' and 'HELP' buttons. Below the header, it says 'User: root Server: ctech59a' and 'SunVTS'. The main content area has a breadcrumb 'Host Machine View > Test Group > Global Test Options'. The title is 'Global Test Options for the host ctech59a'. A note states: 'The test options are set to default values. The options are applied to all the tests. To save modified options, click 'OK' button. To undo changes, click 'Cancel' button.'

Options shown:

- Duration of Testing (min):** A text box containing '0'. Below it, the text 'Specify the duration of testing'.
- Verbose:** Two radio buttons, 'Enable' and 'Disable'. 'Disable' is selected. Below the buttons, the text 'Enabling the Verbose will log the verbose messages to Verbose log file'.
- Duration of Logging (min):** A text box containing '0'. Below it, the text 'Specify the interval of logging the information related to testing status'.

At the bottom, there are 'OK' and 'Cancel' buttons.

2. Set the duration of testing in the **Duration of Testing** field in minutes.

- For Test modes, Online Stress and Component Stress default value for Duration of Testing is set to 0 (infinite).
- For System Exerciser Mode, the Default Duration of Testing is 240 minutes.

3. For the **Verbose** option, select **Enable** if you want to log the Verbose messages or **Disable**, if otherwise.

Verbose messages are the console messages. If verbose is enabled then the console messages from the vtsk are not only displayed on the BUI console, but also logged in the log file. If this box is not checked, all verbose messages are only logged in the log file in `/var/sunvts/logs/sunvts.verbose`.

4. In the **Duration of Logging** option, set the time in minutes after which the information messages should log in the `sunvts.info` file.

For example, if the duration is set to 60 minutes, the testing information messages will be logged in the `sunvts.info` file every 60 minutes.

5. Click **OK** to save the edited global options.

Clicking **Cancel** navigates you back to the Test Group page without saving the edited options. You can also navigate back to the parent page using the links provided at the top of this page.

Safe and Unsafe Options

SunVTS 7.0 PS6 and subsequent compatible releases support a new global option. This option allows you to complete more stressful hardware testing.



Caution – This option needs to be used carefully. To conduct more stressful testing, the test performs operations on the device under test, which can potentially erase any data that is present on the device.

By default all SunVTS tests are data safe. You must decide if you should disable the Safe option: choosing **Disable** performs the more stressful testing. By default, the option is always **Enable** or **Data Safe**.

To provide extra protection, the option will only become active if the `sunvts_options.conf` configuration file is present with the appropriate variable set. All the conditions need to be met before the tool accepts the unsafe testing (the **Disable** option).

▼ To Create a Configuration File for the Unsafe Option

1. Obtain root privileges.

2. Create a configuration file: `/etc/sunvts/conf/sunvts_options.conf`

3. Open the file for editing.

The file takes a variable "name" and a corresponding "value," which will be used by the tool. Both the "name" and the "value" are keywords. The only name and value that are accepted are

- `disk_data_write=Enable`
- `disk_data_write=Disable`

4. Save the file.

5. Start SunVTS.

SunVTS reads the configuration file. In the BUI, the “Global Option” button now has another entry: "Safe Testing". You can set the button to Enable or Disable.

With the Safe Option feature, the Global Test option screen will look similar to [FIGURE 3-18](#).

FIGURE 3-18 Safe Testing Enable or Disable Button

The screenshot displays the SunVTS web interface. At the top, there's a header with 'APPLICATIONS' and 'VERSION' tabs, and 'LOG OUT' and 'HELP' buttons. Below the header, it shows 'User: root Server: ctech59a' and the 'SunVTS' logo. The main content area is titled 'Global Test Options for the host ctech59a'. It includes a note: 'The test options are set to default values. The options are applied to all the tests. To save modified options, click 'OK' button. To undo changes, click 'Cancel' button.' The settings are as follows:

- Duration of Testing (min):** 0. Specify the duration of testing.
- Verbose:** Radio buttons for 'Enable' and 'Disable'. A note states: 'Enabling the Verbose will log the verbose messages to Verbose log file'.
- Duration of Logging (min):** 0. Specify the interval of logging the information related to testing status.
- Safe Testing:** Radio buttons for 'Enable' and 'Disable'. A note states: 'Disabling the parameter will result in data un-safe testing'.

At the bottom, there are 'OK' and 'Cancel' buttons.

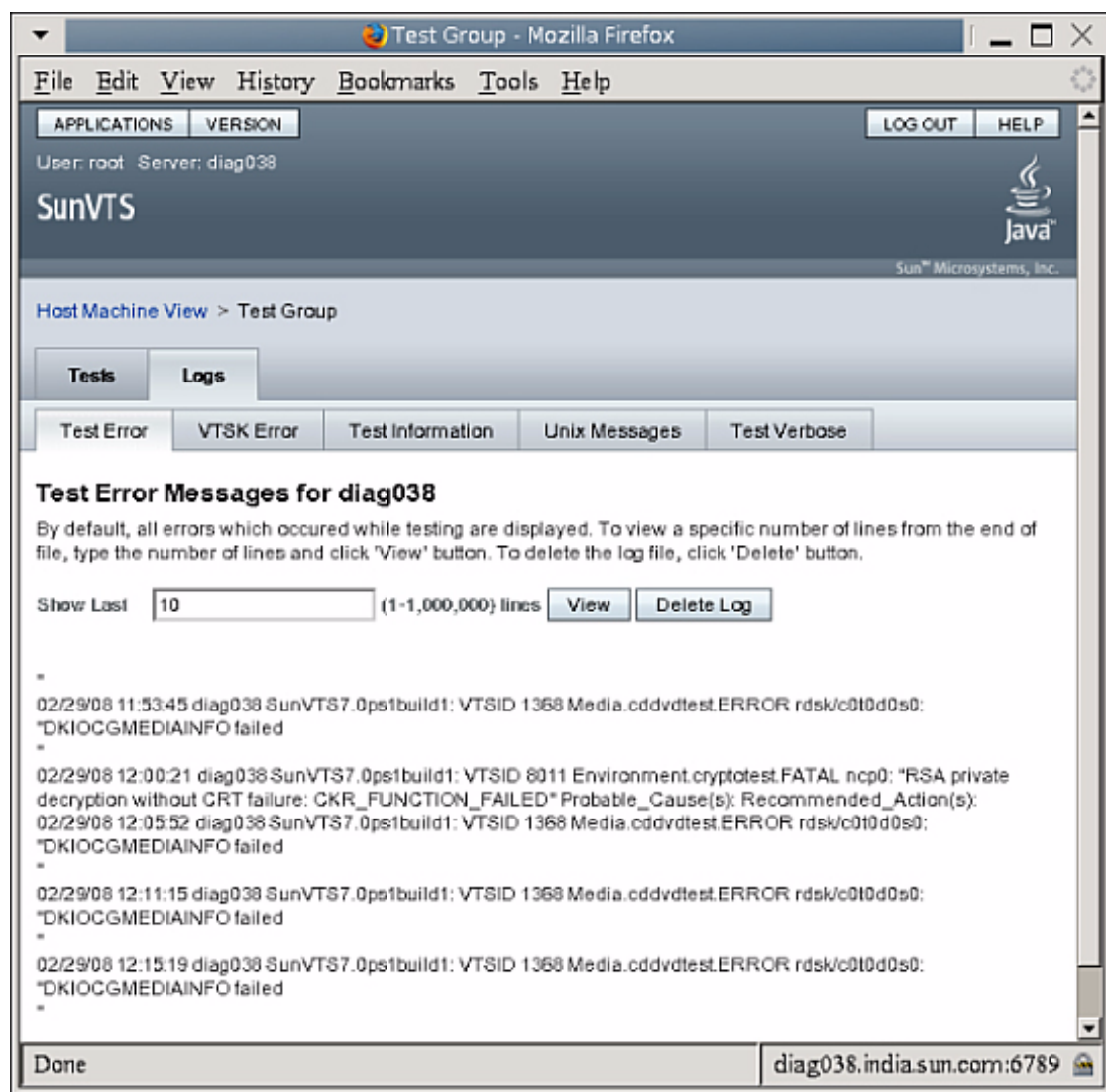
Logs Screen

Logs appears as a tab that can be located next to the Tests tab. Select the Logs tab to view the following five logs. Each of these logs are displayed in five separate tabs.

- Test Error (This log tab is the default display)
- vt sk Error
- Test information
- Unix messages
- Test verbose

Click a specific tab to view related logging information ([FIGURE 3-19](#)).

FIGURE 3-19 Logs Screen Tabs



Types of Logs

Test Error

Click this tab to view the tests related error messages that gets logged to `/var/sunvts/logs/sunvts.err` directory. The test error log contains time-stamped SunVTS test error messages. This file is not created until a SunVTS test failure occurs.

vtsk Error

You can click this tab to view the vtsk related error messages that gets logged to `/var/sunvts/logs/vtsk.err` directory. Vtsk error logs contains time-stamped SunVTS kernel and SunVTS probe errors. SunVTS kernel errors are errors that relate to running SunVTS, and not to testing of devices. This file is not created until SunVTS reports a SunVTS kernel error.

Test Information

Click this tab to view all information about the running tests. Data like the number of test passes, failures, time elapsed, will get collected after a fixed interval of time. Informative messages are generated at the start and stop of the SunVTS testing session and also during intermediate interval notifications. The messages are logged to `/var/sunvts/logs/sunvts.info` directory. This file is not created until a SunVTS test session runs.

Unix Messages

Click this tab to view system generated UNIX messages. It displays all the general Solaris events logged by `syslogd`. The pathname of this log file is `/var/adm/messages`.

Test Verbose Messages

Verbose Messages logs the test messages that can be useful for debugging. The logs displays test messages only when the verbose mode is enabled from Global Options. Verbose message are logged in `/var/sunvts/logs/sunvts.verbose` directory.

Managing Logs

View Logs

You can type the number of lines from the end of log file, that you want to view and then press View button.

Delete Logs

This button clears the test log from the screen and deletes it from the directory. When you click the delete button, a conformation dialog is displayed, to confirm the deletion of the log.

Using the Graphical User Interface

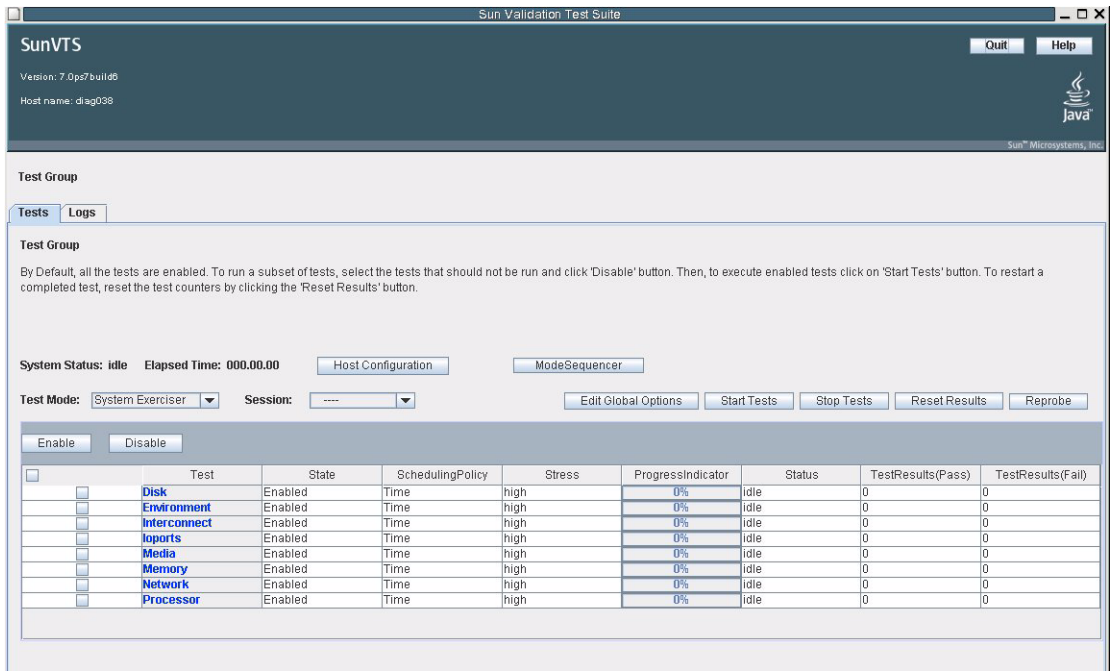
This chapter describes how to run a test session using the SunVTS Graphical User Interface (GUI). Topics Include:

- [“Starting the Graphical User Interface” on page 57](#)
- [“Using the Test Group Screen” on page 58](#)
- [“Using the Logs Screen” on page 72](#)
- [“Using the Auto Mode Sequencer” on page 74](#)

Starting the Graphical User Interface

The SunVTS graphical user interface (GUI) is a Java-based standalone user interface. The Java GUI needs to be started on the same machine under test that has the SunVTS kernel is running. The GUI does not allow remote connection. See [“Starting SunVTS” on page 14](#).

FIGURE 4-1 Main Screen for Graphical User Interface



Using the Test Group Screen

The Test Group screen (FIGURE 4-1) lists the tests that are registered with the host machine. The Tests tab is associated with the Test Group View. The host machine name can be viewed on the heading of the page.

Test Group Table

The Test Group table (FIGURE 4-1) displays the tests registered with the host. You can set the testing modes, schedule the tests, and monitor the testing status of each of the tests.

Also on this page, the feature - session utility can be used to store any specific selection settings. These settings can later be loaded to perform the system testing. You can select the tests to run for testing.

Each row in the table has a test with the default set of test specific values and a progress bar (only for the selected tests once the testing is started).

The progress bar indicates the percentage of the test run based on the selected scheduling criteria. The table includes Actions, Test, State, Scheduling Policy, Stress, Progress, and Status.

Test Table Contents

This section describes the parameters available in setting up a test. Select parameters from the Test Group screen (FIGURE 4-1) and Test Options screen (FIGURE 4-7).

The Test Group screen (FIGURE 4-1) displays the name of the system being viewed for status (santoor in this example). The following items describe the table:

■ Test

Tests can be any of the following: Memory, Disk, Removable Disk, Network, HBA, Graphics, Processor, I/O ports, Interconnect and Media. Click on the test link to set the test options, namely: stress level, scheduling policy (time-based or in terms of test passes) and error limit.

■ State

This column shows the state of the test at a given time (enabled or disabled).

■ Scheduling Policy

On the Test Options screen (FIGURE 4-7), select Test Time or Test Pass based on the testing requirement. In the case of Test Time, specify the time in minutes in the text box (range 0 -99999). For Test Pass, specify the number of passes in the text box (range 0 -99999). By default, Test Time is the scheduling policy and the default value for both (Time or Pass) is zero.

Select Test Pass when time is not a constraint and is being used to reproduce an error on the specific test.

■ Stress

Select low or high (FIGURE 4-7).

■ Progress

The Progress bar in the Test Group table (FIGURE 4-1) indicates the percentage of the test run completed. A tool tip over the progress bar shows the progress of the test in percentage and the current state of the test.

■ Status

The Status column in the Test Group table (FIGURE 4-1) shows the status of the test, either idle, testing, analyzing, not supported, stopping, failed, or completed, along with the passes and error count. Each status item is described as follows.

Idle – A test can be idle for any of the following reasons:

- Initial status of a test before testing.
- SunVTS resets the tests.
- You click the Stop button.
- The global time limit completes and all the tests stop.

Not supported – For the given stress level, a test might not be able to run for any of the following reasons:

- The rules file does not define any task.
- The tests defined under the task (in the rules file) were not able to probe any of the devices.

Analyzing – A test is running after completion of each pass that checks for certain terminating parameters such as max time, max pass, and max error to analyze if it can run for the next pass. The Analyzing status lasts for 2 to 3 seconds.

Testing – The test is currently executing.

Completed – The test has completed its execution. The completion factor is based on the limits set from the scheduling policy, (time or pass limit).

Failed – The test stopped because the test errors reached the maximum error limit.

Waiting for swap – If there is not enough swap available, you can stop a test and schedule the test execution for later.

Stopping – When you press the Stop button, the test status shows the transition from stopping status to idle status.

■ Test Results

The Test Results column displays test results in two subcolumns. The first subcolumn displays the number of test errors and second displays the number of test successes.

Test Group Button Functions

You can perform various functions on the Test Group table ([FIGURE 4-1](#)) using the buttons Enable, Disable, Start Tests, Stop Tests, Edit Global Options, Reset Results and Reprobe. [TABLE 4-1](#) describes the button functions.

TABLE 4-1 Test Group Screen

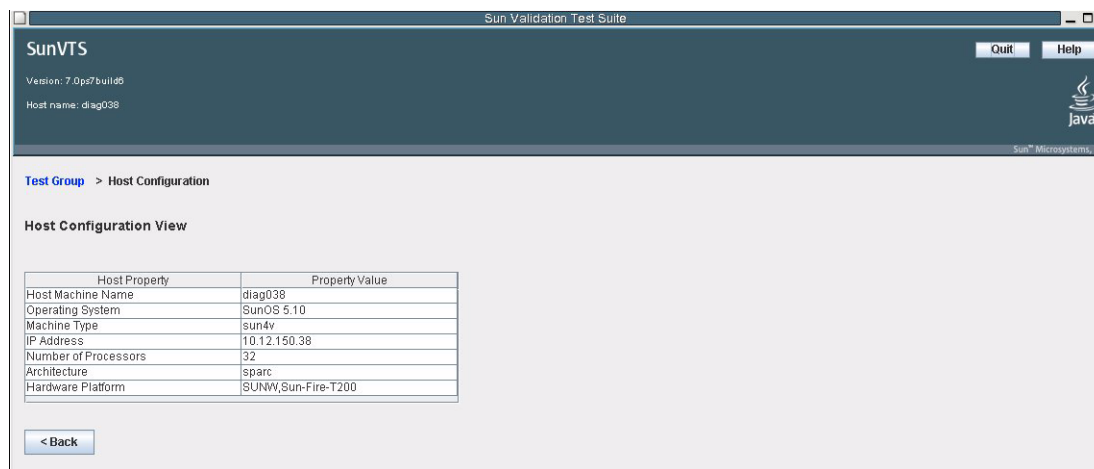
Button	Description
Enable	All tests are enabled. By default, all tests are enabled. To verify which tests are enabled and which are not, look for the state of the tests under the State column corresponding to the test.
Disable	All tests are disabled. To verify if a particular test is disabled, look for the state of the test under the State column corresponding to the test.
Start Tests	Starts all the tests that are enabled. By default, all the tests are enabled. To only run a particular test, disable other tests by clicking the corresponding checkbox and then clicking the Disable button. To start a particular test when other tests are already running, simply select the checkbox for the test and click Enable button.
Stop Tests	Stops all the tests that are started. To stop a particular test that is running, select the checkbox for the test and click the Disable button.
Edit Global Options	Sets options that you can apply to all the tests globally. These options can be used to schedule the total time of the testing session.
Reset Results	Resets the status counters (pass, elapsed time, and error count of a test is reset to zero), and progress bar to an idle state. Before starting any new testing session. Click Reset.
Reprobe	Reprobes the system for devices currently available.

▼ To View the Host Configuration

1. Click on the view host configuration button to fetch the details of the machine under test.
2. Click the View Host configuration button.

The host configuration page is displayed with machine details such as hostname, IP address, platform, operating system, number of CPUs, machine type, and architecture of machine under test ([FIGURE 4-2](#)).

FIGURE 4-2 Host Configuration Window



3. Click the Back button to return to the main page.

You can also use the navigation link at the top of the page to return to the parent page.

Test Modes

Select the Test Mode depending upon the testing requirement. Test Mode is located at the top left corner of the Test Group table. You can select one of the three test modes (Online Stress, System Exerciser, and Component Stress) from the drop down menu. By default the Test Mode is set to System Exerciser mode.

For more information about the test modes, see [“Test Modes” on page 4](#).

Test Session

The Session feature is very useful when you want to automate execution of SunVTS 7.0 with some defined set of options. Instead of setting up the combination of test, global or mode options each time you need to diagnose the host machine, you can create a session file with the combination of options. Once you save your test preferences with session name, you are not required to do any manual setting. The session is saved in XML format. Then only load the session file and the same option settings will be reflected.

Save Session

For selected test mode, you can save chosen tests with configured option values. This feature comes handy when you wish to re-run the tests in the same configuration repeatedly for the same machine or different machine. This is available as a drop-down menu in Tests Screen.

▼ To Save a Test Session

1. **Configure SunVTS for the test session that you want to save.**
2. **Select a Save Session option from Session drop-down menu available in the Test Group screen.**

The Save Session window is displayed ().

3. **Provide a name in the Session Name Text field.**

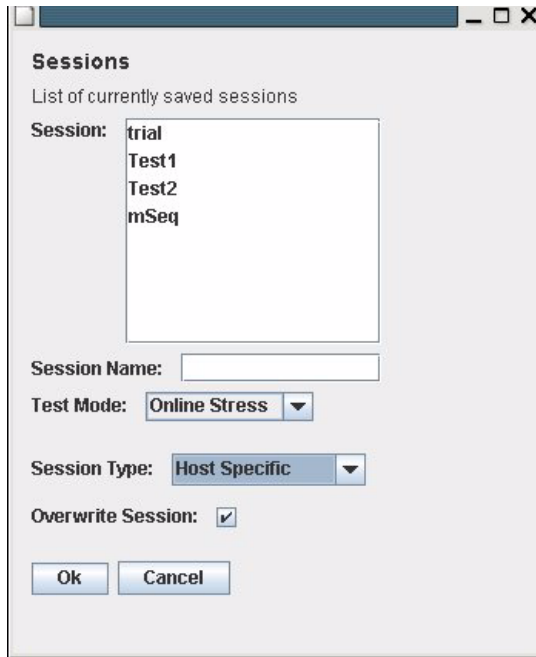
You can also choose existing session name from the list.

Note – Do not use any space for the session name input.

4. **Select a Test Mode as the default active Test Mode for the Session.**
5. **Select Overwrite Session option if you want to overwrite existing session with new.**
6. **Select the type of session, either Generic or Host Specific.**
7. **Click the OK button to save the session.**

The Save Session window is closed, and your session configurations are saved for future use.

FIGURE 4-3 Save Session Window



▼ To List a Session

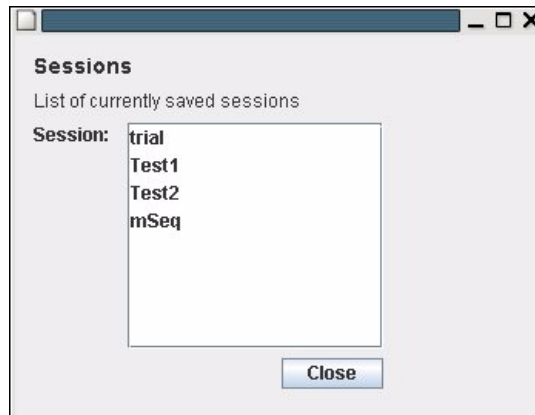
You can view all the sessions existing for the machine. To list sessions:

1. **From the Test Group window, select the List Sessions option from the Session menu.**

The List Sessions window is displayed with the list of all the existing sessions

2. **Click Close to close the List Session Window.**

FIGURE 4-4 List Session Window



▼ To Load a Session

This option allows you to choose desired session. The test configuration and option values for tests execution is set based on selected session. To load sessions:

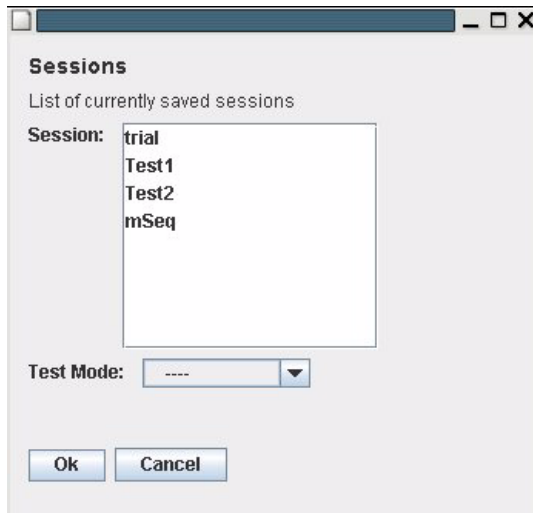
1. **Select a Load Session option from the Session drop-down menu available in the Test Group screen.**

The Load Session window is displayed.

2. **Select an existing session name from the list to load.**
3. **Select a Test Mode as the default Test Mode to load the session.**
4. **Click Ok to Load the session.**

The Load Session window is closed, and the test session configuration is loaded into SunVTS. SunVTS will update and refresh the values on the BUI. The BUI will show the name of the loaded session. You can use this configuration, or modify it, before you start the testing ([FIGURE 4-5](#)).

FIGURE 4-5 Load Session Window



▼ To Delete a Session

If you no longer need a specific test configuration, you can delete the session associated with it. To delete a session:

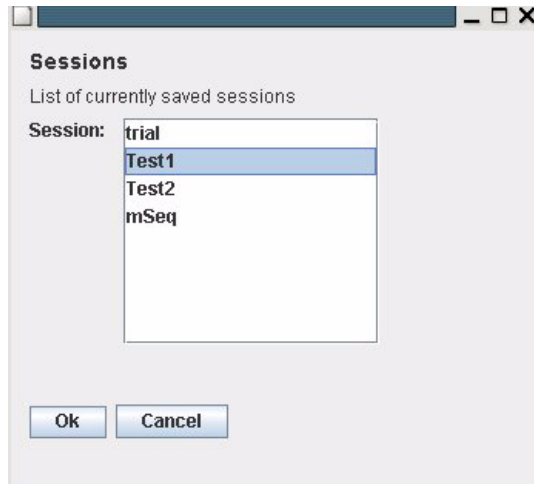
1. **Select Delete Sessions option from the Session drop-down menu available in the Test Group screen.**

The Delete Session window is displayed with the list of all existing sessions.

2. **Select the session that you want to delete from the List of Sessions.**
3. **Click OK to delete the selected Session.**

The Delete Session window is closed, and selected session will be deleted.

FIGURE 4-6 Delete Session Window



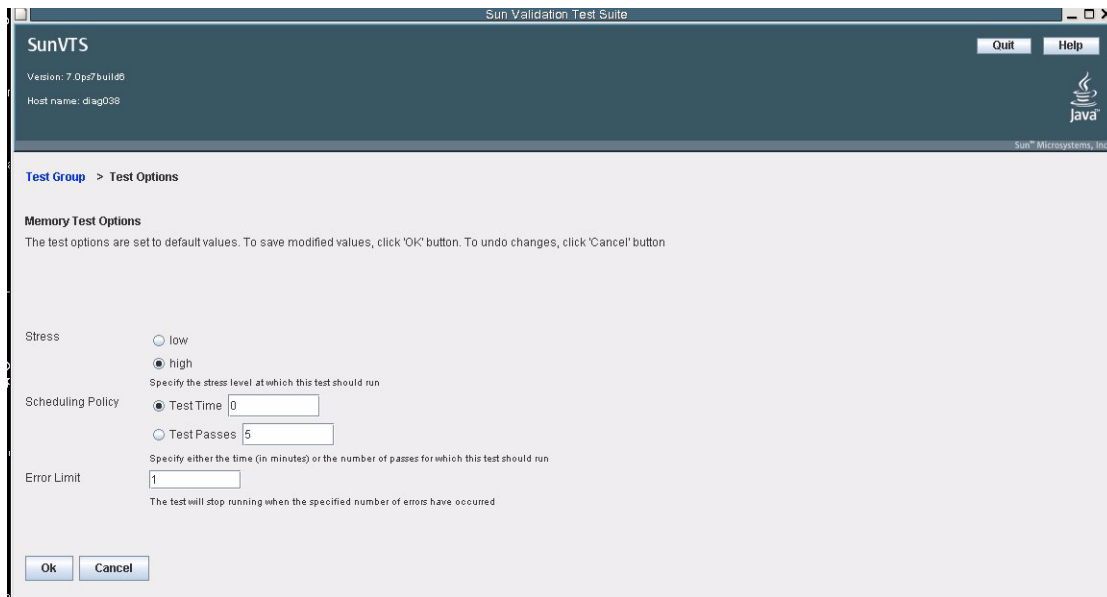
Reset Session

If you loaded or saved a session but then decide to run the tests with the default test configuration, you can use this option. This clears the values and settings done by the session and loads default parameters.

Test Options Screen

The Test Options screen ([FIGURE 4-7](#)) appears when you click the Test Name from the Test Group table in the Test column.

FIGURE 4-7 Test Options Screen



▼ To Change Test Options

1. Specify the stress level - low or high, depending upon your requirement (FIGURE 4-7).

The default value of stress are based on the configuration of the system. If the system has a small memory configuration, then the default stress is set to “low” level. Otherwise, the default stress is “high” level. SunVTS considers a system as a small memory configuration if it has less than or equal to 2GB of memory (for x86), and less than or equal to 4GB (for SPARC).

2. Choose the Scheduling Policy between Test Time or Test Passes.

Depending on the selection, provide the time in minutes or number of passes.

- Online Stress Default - Scheduling Policy is Test Pass, and the default Test Pass is 1. If the scheduling policy is changed to Test Time, the default Test Time will be 240 minutes.
- System Exerciser Default - the Scheduling Policy is Test Time, and the default Test Time is 0 (or infinite). If the scheduling policy is changed to Test Pass, the default Test Pass is 5.
- Component Stress Default - Scheduling Policy is Test Pass, and the default Test Pass is 5. If the scheduling policy is changed to Test Time, the default Test Time will be 240 minutes.

3. Choose a Specific Pass Level.

You can choose a specific pass level and only the test related to that specific pass level will run. The valid range of pass levels are listed in the enclosed bracket next to the specific pass level tag. By default, the value of Specific Pass level is 0 and means that no specific pass level has been selected.

4. Choose an Error Limit.

Specify the number of errors beyond which the test should stop executing. By default, the Error Limit is set to 1. An error limit of 0 would mean that the test continues to run with no limits to error count.

5. Choose specific test device options (optional).

Some Tests export specific options that allow you to select the test devices and modify the device options.

a. **Select the check box corresponding to the device and click the “enable” or “disable” button to select/de-select the device. By default, all the devices will be enabled or disabled.**

b. **To change the device options, click on the device link.**

The test option selection window is displayed.

c. **Edit the test options.**

6. Click OK to confirm the changes and to return to the parent page.

The Cancel button cancels the changes and returns to the parent page.

Note – Setting a Test Time to 0 (zero) or Test Pass to 0 (zero) indicates that the test must run for an infinite time or infinite pass respectively. In such a situation the progress bar will not show significant change. You must specify the Test Time or Test Pass with an integer value to see correct indication of the progress bar.

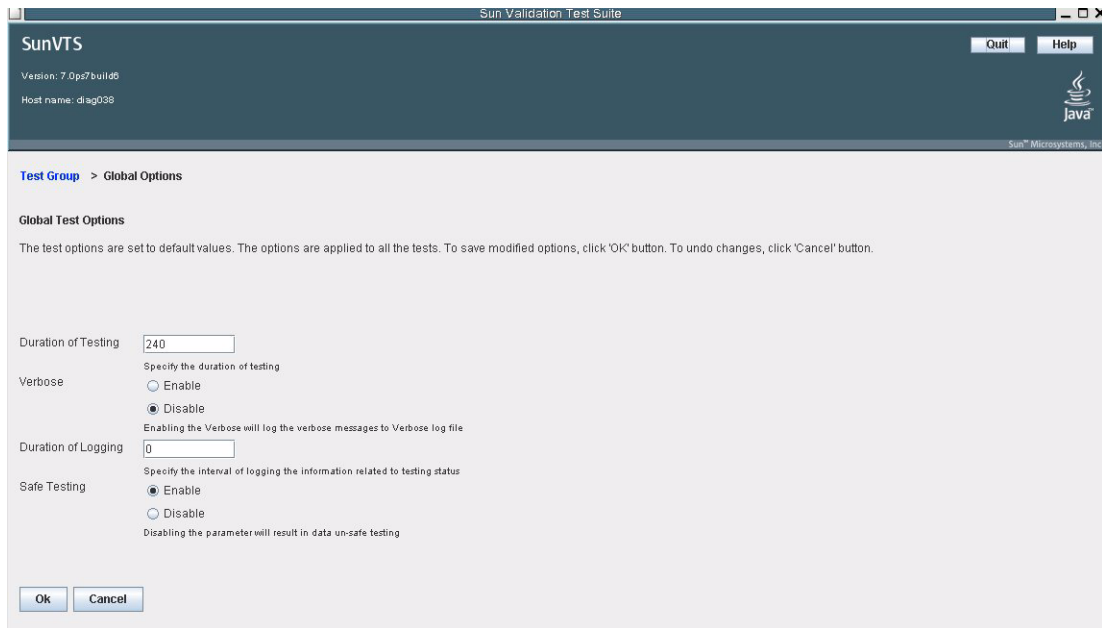
▼ To Set Global Options

There are certain options that can be applied to all the tests globally. These options can be used to schedule the total time of the testing session. To edit the Global options:

1. Click on the Edit Global Options button on the Test Group screen.

The Global Test Options screen is displayed (FIGURE 4-8).

FIGURE 4-8 Setting Global Options



2. Set the duration of testing in the Duration of Testing field in minutes.

- For Test modes, Online Stress and Component Stress default value for Duration of Testing is set to 0 (infinite).
- For System Exerciser Mode, the Default Duration of Testing is 240 minutes.

3. For the Verbose option, select Enable if you want to log the Verbose messages or Disable, if otherwise.

Verbose messages are the console messages. If verbose is enabled then the console messages from the vtsk are not only displayed on the BUI console, but also logged in the log file. If this box is not checked, all verbose messages are only logged in the log file in `/var/sunvts/logs/sunvts.verbose`.

4. In the Duration of Logging option, set the time in minutes after which the information messages should log in the `sunvts.info` file.

For example, if the duration is set to 60 minutes, the testing information messages will be logged in the `sunvts.info` file every 60mins.

5. Click OK to save the edited global options.

Clicking Cancel navigates you back to the Test Group page without saving the edited options. You can also navigate back to the parent page using the links provided at the top of this page.

Safe and Unsafe Options

SunVTS 7.0 Patch Set 6 (PS6) and subsequent compatible releases support a new global option. This option allows you to perform more stressful hardware testing.



Caution – Use the unsafe option carefully. To conduct more stressful testing, the test performs operations on the device under test and erases data that is present on the device.

By default all SunVTS tests are data safe. You must decide whether to choose to Disable the Safe option: choosing Disable performs the more stressful testing. By default, the option is always Enable or Data Safe.

To provide extra protection, the option will only become active if the `sunvts_options.conf` configuration file is present with the appropriate variable set. All the conditions need to be met before the tool accepts the unsafe testing (the Disable option).

The details on creation of the configuration file and its setting are mentioned below:

▼ To Create a Configuration File for the Unsafe Option

1. Obtain root privileges.

2. Create a configuration file: `/etc/sunvts/conf/sunvts_options.conf`

3. Open the file for editing.

The file takes a variable "name" and a corresponding "value," which will be used by the tool. Both the "name" and the "value" are keywords. The only name and value that are accepted are:

- `disk_data_write=Enable`
- `disk_data_write=Disable`

4. Save the file.

5. Start SunVTS.

SunVTS will now read the configuration file. On the GUI, the "Global Option" button now has another entry: "Safe Testing". You can set the button to Enable or Disable.

With the Safe Option feature, the Global Test option screen will look similar to [FIGURE 4-8](#).

Using the Logs Screen

Logs appears as a tab that can be located next to the Tests tab. Select the Logs tab to view the following five logs. Each of these logs are displayed in five separate tabs.

- Test Error (This log tab is the default display)
- Vtsk Error
- Test information
- Unix messages
- Test verbose

Click a specific tab to view related logging information ([FIGURE 4-9](#)).

FIGURE 4-9 Logs Screen



Types of Logs

Test Error

Click this tab to view the tests related error messages that gets logged to `/var/sunvts/logs/sunvts.err` directory. The test error logs contains time-stamped SunVTS test error messages. This file is not created until a SunVTS test failure occurs.

Vtsk Error

You can click this tab to view the vtsk related error messages that gets logged to `/var/sunvts/logs/vtsk.err` directory. Vtsk error logs contains time-stamped SunVTS kernel and SunVTS probe errors. SunVTS kernel errors are errors that relate to running SunVTS, and not to testing of devices. This file is not created until SunVTS reports a SunVTS kernel error.

Test Information

Clicking this tab will provide all information about the running tests. Data like number of test passes, failures, time elapsed, will get collected after fixed interval of time. Informative messages are generated at the start and stop of SunVTS testing session and also during intermediate interval notifications. The messages will be logged to `/var/sunvts/logs/sunvts.info` directory. This file is not created until a SunVTS test session runs

Unix Messages

Click this tab to view system generated UNIX messages. It displays all the general Solaris events logged by `syslogd`. The pathname of this log file is `/var/adm/messages`.

Test Verbose Messages

Verbose Messages logs the test messages that can be useful for debugging. The logs displays test messages only when the verbose mode is enabled from Global Options. Verbose message are logged in `/var/sunvts/logs/sunvts.verbose` directory.

Managing Logs

View Logs

- **Type the number of lines from the end of log file that you want to view and press the View button.**

Delete Logs

- This button clears the test log from the screen as well as delete it from the directory. Click the delete button.

A conformation dialog is displayed to confirm the deletion of the log.

Using the Auto Mode Sequencer

Under normal circumstances, you run only one mode at a time in a session. Once testing with a mode is over, you must switch the mode in the same session and start the testing if it is required. See [“Saving a Test Session Configuration for Repeated Use \(Session Files\)”](#) on page 90.

Use the Auto Mode Sequencer if a testing session requires running different modes automatically. You can select any of the three modes for the first, second and third sequences. It can be the same mode for all three sequences. You can specify one of the modes and leave the other two sequences as None. This will result in running only one mode for loop number of times.

Note – At least one sequence must have a mode specified. If no modes are selected for the sequence, the sequencer will behave as if it is set to off, even if the sequencer option is set to “on”.

When testing is started with the Auto Mode Sequencer, if the testing incurs an error in any of the modes in the sequence, the Auto Mode sequencer will halt and will not switch to the next mode in the sequence. This allows you to analyze the test error. If the testing had continued without halting, the error might occur without notice.

▼ To Edit Auto Mode Sequencer Options

Use the sequencer to determine the sequence of tests in auto mode. By default, the Sequencer is switched off (set to Disable).

1. Set the Sequencer option to Enable.

2. Select the modes.

- First - Choose one of the three modes that will run first. By default, the option is set to None.
- Second - Choose one of the three values that will run second. By default, the option is set to None.

- Third - Choose one of the three values that will run third. By default, the option is set to None.

Loop

Loop determines the number of times the sequence will run.

1. Specify a numerical value for this option.

The default value is 1. The maximum value is 99999.

2. Click OK to apply the changes.

FIGURE 4-10 Auto Mode Sequencer Screen

Test Group > Auto Mode Sequencer

Auto Mode Sequencer

The options are set to default values. To save modified options, click 'OK' button. To undo changes, click 'Cancel' button.

Sequencer	<input type="radio"/> Enable <input checked="" type="radio"/> Disable Enabling the Sequencer will switch on the mode sequencer feature
First	<input checked="" type="radio"/> None <input type="radio"/> Online Stress <input type="radio"/> System Exerciser <input type="radio"/> Component Stress Select the first mode to run in the sequence
Second	<input checked="" type="radio"/> None <input type="radio"/> Online Stress <input type="radio"/> System Exerciser <input type="radio"/> Component Stress Select the second mode to run in the sequence
Third	<input checked="" type="radio"/> None <input type="radio"/> Online Stress <input type="radio"/> System Exerciser <input type="radio"/> Component Stress Select the third mode to run in the sequence
Loop	<input type="text" value="1"/> that the sequence should keep repeating

Using the SunVTS TTY User Interface

This chapter describes how to run a test session using the SunVTS TTY user interface (UI). Topics include:

- [“Navigating the SunVTS TTY User Interface” on page 78](#)
- [“Configuring SunVTS for a Test Session” on page 80](#)
- [“Reviewing and Resetting Test Session Results” on page 87](#)
- [“Additional SunVTS TTY User Interface Features” on page 89](#)

Note – SunVTS must first be started before selecting a user interface. See [“Starting SunVTS” on page 14](#).

Starting the SunVTS TTY User Interface

The SunVTS TTY UI is an ASCII-based, screen-oriented interface that does not require a graphical window environment. Most of the control and monitoring options that are available in the SunVTS graphical interfaces are also available in the TTY interface.

You can use the TTY interface to run SunVTS from a serial terminal, or through a graphical window tool such as a shell tool, command tool, or terminal window. [FIGURE 5-1](#) shows the TTY main window. (See [“Navigating the SunVTS TTY User Interface” on page 78](#) for a description of the keyboard commands used for the TTY interface.)

FIGURE 5-1 TTY Main Window

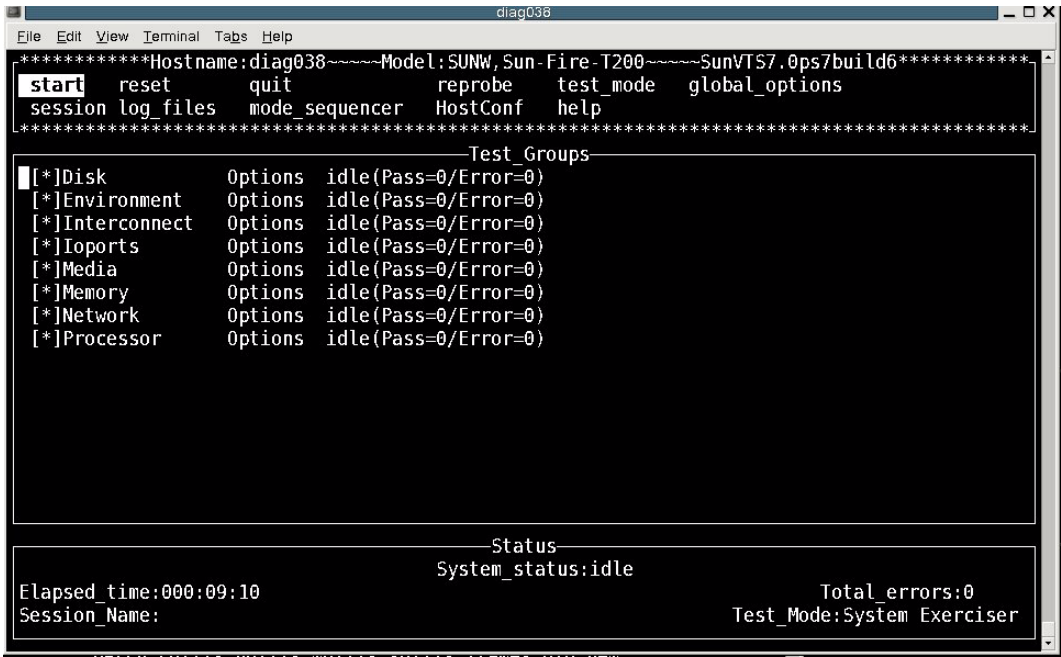


TABLE 5-1 describes items in the TTY main window.

TABLE 5-1 SunVTS TTY Main Window

Window item	Description
Control panel	Provides access to SunVTS controls.
Test_Groups panel	Shows the testable devices (categorized in groups) and provides access to their test options.
Status panel	Displays SunVTS test status information.

Navigating the SunVTS TTY User Interface

Use keyboard commands instead of the mouse to control SunVTS in the TTY UI. TABLE 5-2 lists the keyboard commands for the TTY UI.

TABLE 5-2 Keyboard Commands for SunVTS TTY User Interface

Key	Description
Tab	Moves you from one panel to another. For example, if you are working in the Control panel and press the Tab key, the focus (or highlight) shifts to the Test Groups panel. Highlighted panels are bordered by asterisks (*).
Arrow keys	Moves between selections within a panel.
Return	Displays a menu. Selects and applies a choice or a command in a menu.
Spacebar	Selects or deselects a checkbox: [*] = select [] = deselect
Backspace key	Deletes text in a text field.
Escape	Discards a pop-up menu or window.
Control-F	Scrolls forward in a scrollable window.
Control-B	Scrolls backward in a scrollable window.
Control-X	Quits the TTY user interface but leaves the SunVTS kernel running.
Control-L	Refreshes the TTY window.

Configuring SunVTS for a Test Session

This section describes how to configure SunVTS for a test session.

Identifying the SunVTS Status

View the TTY main window Status panel for the current state of SunVTS. The following describes the System Status options:

- **Idle** – A test can be idle for any of the following reasons:
 - Initial state of a test.
 - When SunVTS resets the tests.
 - When you click the Stop button.
 - When the global time limit completes and all the tests stop.
- **Not supported** – For the given stress level, a test might not be able to run for any of the following reasons:
 - The rules file does not define any task.
 - The tests defined under the task (in the rules file) were not able to probe any of the devices.
 - When you click the Stop button.
 - When the global time limit completes and all the tests stop.
- **Analyzing** – A test after completion of each pass checks for certain terminating parameters such as max time, max pass, and max error to analyze if it can run for the next pass. The Analyzing status lasts for two to three seconds.
- **Testing** – This status indicates that the test is currently executing.
- **Completed** – This status indicates that the test has completed its execution. The completion factor is based on the limits set from the scheduling policy, that is, time or pass limit.
- **Failed** – This status indicates that the test has been stopped because the test errors have reached the maximum error limit.
- **Waiting for swap** – When there is not enough swap availability, you can stop a test and schedule the test execution for later.
- **Stopping** – When you press the Stop button, the test status shows the transition from stopping status to idle state.

Selecting a Test Mode

The test modes consist of Online Stress mode, System Exerciser mode, and Component Stress mode.

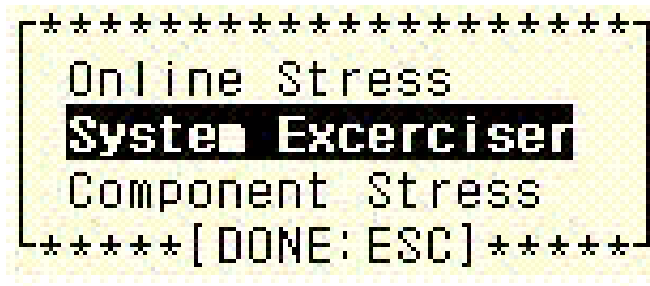
The test modes are described in [Chapter 1, “Test Modes”](#) on page 4.

By default, the test mode is set to the System Exerciser mode.

To select a test mode:

1. Use the **Tab** key to move to the **Control panel** (if needed) in the TTY main window ([FIGURE 5-1](#)).
2. Use the **arrow keys** to select (highlight) `test_mode` and press **Return**.
The `test_mode` menu appears and displays the `test_mode` options ([FIGURE 5-2](#)).

FIGURE 5-2 Test Mode



3. Use the **arrow keys** to select either the **System Exerciser**, **Component Stress**, or **Online Stress** mode and press **Return**.

Changing the Global Options

When global options are changed, these options remain applied for the entire test session. See [“Using the SunVTS Browser User Interface”](#) on page 25 for more information about these options.

To change the global options:

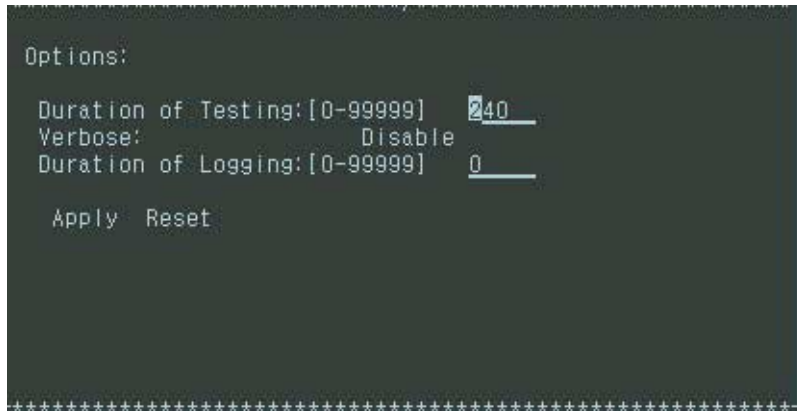
1. Use the **Tab** key to move to the **Control panel** (if needed) in the TTY main window ([FIGURE 5-1](#)).
2. Use the **arrow keys** to select (highlight) `global_options` and press **Return**.
The `global_options` menu appears and displays options ([FIGURE 5-3](#)).

3. In the **Duration of Testing** field, enter the numerical value for the desired test duration and press **Return** (FIGURE 5-3).

This sets all the global options changed.

4. Use the **arrow key** and select the **Verbose** option and press **Return** (FIGURE 5-3).

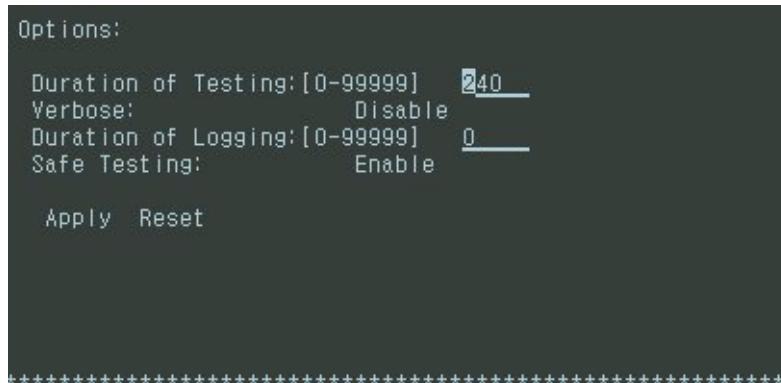
FIGURE 5-3 Global Option Menu – Changing the Global Options



5. Enable or Disable the **Verbose** option using the **arrow keys** to select your choice and press **Return**.
6. In the **Duration of logging** field, enter the duration in minutes.
The testing information messages are logged at regular intervals.

Note – To work with the Safe Option, see [“Safe and Unsafe Options”](#) on page 51. If the feature is enabled, the global Options screen in TTY will be as shown in [FIGURE 5-4](#).

FIGURE 5-4 Duration of Logging



Changing Test Specific Options

Test specific options are specific to individual tests.

To change specific test options:

1. Use the **Tab** key to move to the test groups menu (if needed) in the TTY main window ([FIGURE 5-1](#)).
2. Use the arrow keys to move to individual tests ([FIGURE 5-5](#)).
3. To enable or disable a particular test, highlight the test using arrow keys.
4. Use the arrow keys to highlight the [] (brackets) next to the test that you want to select or deselect.
5. Press the space bar to toggle the asterisk.
6. Move the cursor using arrow keys to the options and press return.

FIGURE 5-5 Selecting Specific Test Options



7. Change the values.

To change the value of Stress to a high or a low value:

- a. Move the cursor to the default value in the Stress field and press Return.**

A display appears showing two entries, low and high (FIGURE 5-6).

- b. Select the desired stress level and press Return.**

To change the Scheduling Policy:

- a. Move the cursor to the Scheduling Policy field and press Return.**

A display appears showing two entries, Test time or Test passes base (FIGURE 5-7).

- b. Select the desired option and press Return.**

8. Change the device specific options:

- a. Move the cursor to Device Options and Click to view the choices (FIGURE 5-8).**

A list of exported devices is displayed.

- b. Move the cursor to Options and press Return.**

- c. Select or add the desired entries for the displayed options.**

- d. Move the cursor to the Apply button and press Return.**

The previous screen is displayed.

- e. Move the cursor to the Apply button and press Return to apply the options.**

FIGURE 5-6 Changing the Stress Option

```
Options:

Stress:                                high
Scheduling Policy:                      [*****]me
Test Time:[0-99999]                    [low]
Test Passes:[0-99999]                  high
Error Limit:[0-99999]                  [*****]

Apply  Reset
```

FIGURE 5-7 Changing the Scheduling Policy

```
Options:

Stress:                                high
Scheduling Policy:                      Test Time
Test Time:[0-99999]                    [*****]
Test Passes:[0-99999]                  [Test Time]
Error Limit:[0-99999]                  Test Passes
                                           [*****]

Apply  Reset
```

FIGURE 5-8 Displaying Device Options

```
*****Device Options*****
[*]c0t1d0p0  Options
[*]c0t0d0    Options
*****[DONE:RETURN]*****
```

FIGURE 5-9 Changing Device Options

```
*****Device Options*****

Options:

Iostat:                                Disable
Write Threshold(KB/s): 0
Read Threshold(KB/s): 0

Apply  Reset
```

Enabling or Disabling Tests

1. Use the **tab** key to go to **Test_Groups** window in the TTY main window (FIGURE 5-1).
2. Use the arrow keys to go to the **[*]/[]** (brackets) button.
3. Use the arrow keys to go to the test to be enabled or disabled.
4. Press the **space bar** to enable or disable the test.

Running and Monitoring a Test Session

This section describes how to run, monitor and stop test sessions.

Starting the Test Session

1. Use the **tab** key to move to the **Control** panel in the TTY main window (FIGURE 5-1).
2. Use the arrow keys to select (highlight) **start** and press **Return**.

The test session then begins. Depending on how you configured the options in the **Test_Execution** menus, testing continues until one of the following occurs:

- Testing has reached the **Max_Passes** value. The default is to run indefinite test passes.
- Testing has reached the **Max_Time** value. The default is to run indefinitely.
- You select **Stop** in the **Control** panel.

Monitoring Tests

Monitor the overall test session status by watching the **Status** panel. The **Status** panel indicates that a test session is running by displaying testing for the **System_status**. The field **elapsed_time** will show how much time has elapsed since tests were started. **Total_errors** will show how many errors were encountered since the tests began execution.

Monitor specific tests by watching the **Status** panel. An asterisk displayed next to each device indicates which test (for example, **Processor test**) is currently under test.

Stopping the Test Session

1. Use the Tab key to move to the Control panel in the TTY main window (FIGURE 5-1).
2. Use the arrow keys to highlight `stop`, and press Return.
3. Press Esc to dismiss the confirmation menu.

The test session stops. The test session is being stopped and the system status displays “stopping”. The main window shows a disabled stop button as `<stop>`. When the tests completely stop, the “start” button appears.

Reviewing and Resetting Test Session Results

This section shows how to review logs, interpret SunVTS messages, and reset test session results.

Reviewing Logs

SunVTS provides access to four log files:

- **SunVTS test error log** – contains time-stamped SunVTS test error messages. The log file pathname is `/var/sunvts/logs/sunvts.err`. This file is not created until a SunVTS test failure occurs.
- **SunVTS kernel error log** – contains time-stamped SunVTS kernel and SunVTS probe errors. SunVTS kernel errors are errors that relate to running SunVTS, and not to testing of devices. The log file pathname is `/var/sunvts/logs/vtsk.err`. This file is not created until SunVTS reports a SunVTS kernel error.
- **SunVTS information log** – contains informative messages that are generated when you start and stop SunVTS test sessions. The log file pathname is `/var/sunvts/logs/sunvts.info`. This file is not created until a SunVTS test session runs.
- **Solaris system message log** – is a log of all the general Solaris events logged by `syslogd`. The pathname of this log file is `/var/adm/messages`.

- **SunVTS Verbose log** – Verbose messages are the test progression informative messages that are logged to the log file in `/var/sunvts/logs/sunvts.verbose`. These messages are logged only when the verbose mode is enabled and helps you continuously monitor what is happening during the testing.

To review log files:

1. Use the tab keys to move to the Control panel in the TTY main window (FIGURE 5-1).
2. Use the arrow keys to highlight `log_files`.

The `log_files` menu appears (FIGURE 5-10).

FIGURE 5-10 Log File Menu

```
log_files    HostConf
[*****log_files*****]
  Verbose
  Test Error
  VTS Kernel Error
  Information
  Unix Msgs
[***** [DONE:ESC] *****]
```

3. Use the arrow keys to select one of the log files and press Return (FIGURE 5-10).

FIGURE 5-11 Log File Display Option

```

*****
[Display]
  Remove
  Print
[ [DONE:ESC] ]

```

4. Use the arrow keys to select one of the following and press Return (FIGURE 5-11):
 - Display – the content of the log file appears.

Note – SunVTS uses your default editor to display the log file. In many cases, the default test editor is `vi` and you can use the standard `vi` commands to view the content. However, the file is opened in read-only mode, so you cannot do anything to alter the content. Type `q` to quit and return to the main TTY window.

- Remove – the content of the log file is deleted.
- Print – the print menu appears. Specify the name of the printer in this menu, highlight Apply, and press return. The content of the log file is sent to the printer.

Note – Take the length of the log file into consideration before you print it.

5. Press `Esc` to close the `log_file` menu.

Resetting Test Session Results

Resetting test sessions resets all the tests pass, error, and time limit counts to zero. Reset is only possible if the SunVTS System Status is idle. If the system status is not idle, then an error is returned.

To reset a test session:

1. Use the `tab` key to move to the Control panel in the TTY main window (FIGURE 5-1).
2. Use the arrow keys to highlight `reset` and press Return.

The information in the Status panel is reset.

Note – This does *not* reset your test options.

Additional SunVTS TTY User Interface Features

This section describes additional SunVTS features using the SunVTS TTY user interface.

Saving a Test Session Configuration for Repeated Use (Session Files)

You can use the Session Files feature to save the current set of selected tests and test options for reuse. This is a convenient feature when you plan to use the same test session configuration repeatedly.

The configuration information is saved in a file name of your choice in the `/var/sunvts/sessions` directory.

▼ To Create a Session File

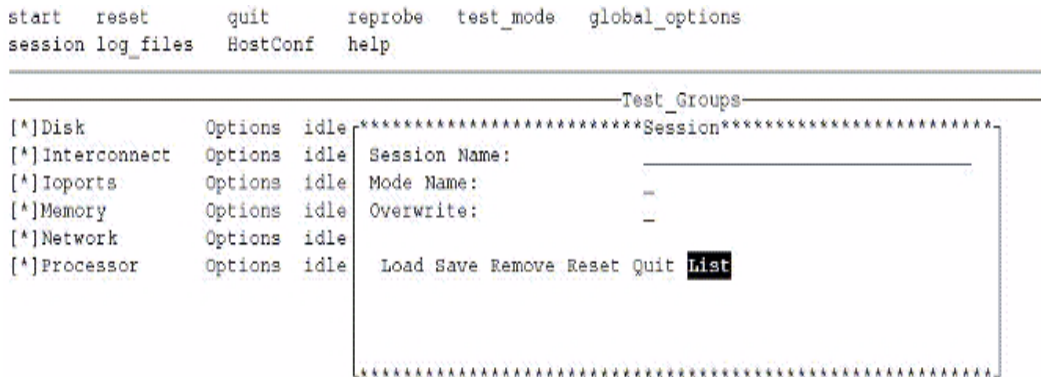
1. **Configure SunVTS for the test session that you want to save.**

See [“Configuring SunVTS for a Test Session”](#) on page 80.

2. **Select (highlight) session in the Control panel in the TTY main window (FIGURE 5-1), and press Return.**

The Session Files menu appears (FIGURE 5-12).

FIGURE 5-12 Sessions File Menu



3. **Select a session file name in the Session file field.**
4. **Enter the mode name.**
s for System Exerciser, **c** for Component Stress or **o** for Online Stress modes. or leave it blank to save it with the default mode of the current test session.
5. **Enter y for yes or n for no if you are overwriting and saving with the same name as an existing session.**

6. Move the cursor to save the menu item on the Session display and press Return to save the session.

Note – To access Session Files from the TTY UI, the Session File name must be 30 characters or less-the TTY UI has a maximum of 30 characters for input.

▼ To Load a Session File

Note – Loading a Session file that was created on another system is permitted, but you must make sure that the configuration is valid for the system you plan to test.

1. Select (highlight) session in the Control panel in the TTY main window (FIGURE 5-1), and press Return.

The Session Files menu appears (FIGURE 5-12).

2. Select (highlight) List and press Return.

The list of available session files appears (FIGURE 5-13).

FIGURE 5-13 Sessions File – Sessions Displayed

```

Session
-----
Session Name:
Mode Name:
Overwrite:

Load Save Remove Reset Quit List
*****Sessions*****
ab    db    aj
aj31  bosc  ajay

*****[DONE:ESC]*****
```

Note – To access session files from the TTY UI, the session file name must be 30 characters or less. The TTY UI has a maximum of 30 characters for input.

3. Select (highlight) a session file from the list, and press Return.
4. Select (highlight) Load, and press Return.

The test session configuration is loaded into SunVTS. You can use this configuration, or modify it before you start the test session.

▼ To Remove a Session File

1. **Select (highlight) session in the Control panel in the TTY main window (FIGURE 5-1), and press Return.**

The Session files menu appears (FIGURE 5-13).

2. **Select (highlight) List and press Return.**

The list of available session files are displayed.

3. **Select (highlight) the session file from the list that you want to delete, and press Return.**

Note – To access session files from the TTY UI, the session file name must be 30 characters or less. The TTY UI has a maximum of 30 characters for input.

4. **Select (highlight) Remove, and press Return.**

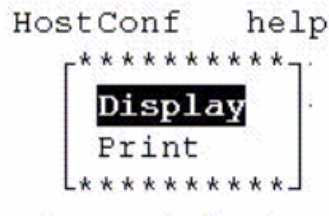
Viewing Host System Configuration

To view a host system configuration:

1. **Use the tab key to move to the main Control panel in the TTY main window (FIGURE 5-1).**
2. **Use the arrow keys to select (highlight) the HostConf button and press Return.**

This displays the option to display or print the System Configuration of the machine under test (FIGURE 5-14).

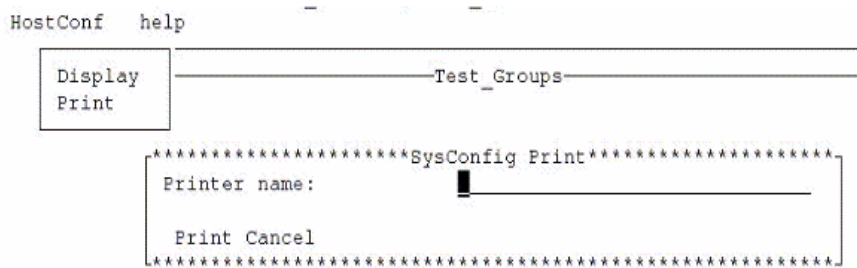
FIGURE 5-14 Host Configuration



3. Select **Display**, and press **Return**, to display the system configuration.
4. Select **Print**, and press **Return**, to view the host configuration print display (FIGURE 5-15).

Enter the printer name and press the **Print** button to print the System Configuration.

FIGURE 5-15 Host Configuration – Print Display



Reprobing the Tests

Reprobing is a feature provided in SunVTS which reprobes and registers the devices either of which are added under testing to the SunVTS. This feature also removes the device from the SunVTS listing if the device has been removed from the system.

To reprobe a test:

1. Use the **tab** key to move to the **Control** panel in the TTY main window (FIGURE 5-1).
2. Use the **arrow** keys to select **reprobe** and press **Return**.
The **reprobe** screen appears (FIGURE 5-16).
3. Press **Ok** to start reprobing.

```
reprobe    test_mode    global_options
help
```

```
*****
OK to reprobe system?
cancel ok***]
*****
```

To access help and the Release Table:

- FIGURE 5-17** Help Option

```
help
*****
vtstty help
Release Table
*****
```


Using the Auto Mode Sequencer

The Auto Mode Sequencer will only function when this option is enabled. By default, the Sequencer is switched off, or set to Disable.

Under normal circumstances, you run only one mode at a time in a session. Once testing with a mode is over, you must switch the mode in the same session and start the testing if it is required. See [“Saving a Test Session Configuration for Repeated Use \(Session Files\)” on page 90](#).

Use the Auto Mode Sequencer if a testing session requires running different modes automatically. You can select any of the three modes for the first, second and third sequences. It can be the same mode for all three sequences. You can specify one of the modes and leave the other two sequences as None. This will result in running only one mode for loop number of times.

Note – At least one sequence must have a mode specified. If no modes are selected for the sequence, the sequencer will behave as if it is set to off, even if the sequencer option is set to “on”.

When testing is started with the Auto Mode Sequencer, if the testing incurs an error in any of the modes in the sequence, the Auto Mode sequencer will halt and will not switch to the next mode in the sequence. This allows you to analyze the test error. If the testing had continued without halting, the error might occur without notice.

▼ To Edit Auto Mode Sequencer Options

Use the sequencer to determine the sequence of tests in auto mode. By default, the Sequencer is switched off (set to Disable).

Sequencer

1. Set the Sequencer option to Enable ().

2. Select the modes.

- First - Choose one of the three modes that will run first. By default, the option is set to None.
- Second - Choose one of the three values that will run second. By default, the option is set to None.
- Third - Choose one of the three values that will run third. By default, the option is set to None.

Loop

Loop determines the number of times the sequence will run.

1. Specify a numerical value for this option.

The default value is 1. The Maximum is 99999.

FIGURE 5-18 Auto Mode Sequencer Screen in the TTY Interface

```
-----Hostname:diag038-----Model:SUNW,Sun-Fire-T200-----SunVTS7.0ps7build6-----
start   reset   quit      reprobe   test_mode  global_options
session log_files  mode_sequencer HostConf  help

[+]Disk
[+]Envir
[+]Inter
[+]Iopor
[+]Media
[+]Memor
[+]Netwo
[+]Proce

*****Mode_sequencer*****
Options:
Sequencer:  Disable
First:      None
Second:     None
Third:      None
Loop:[1-99999]  1
Apply  Reset

*****
-----Status-----
System_status: idle
Elapsed_time:000:00:00      Total_errors:0
Session_Name:              Test_Mode:System Exerciser
```

2. Select Apply to set all the changes.

Quitting the TTY User Interface

The Quit option provides a way to exit out of the terminal user interface. This can be done when the testing is not in progress and system status is idle.

To quit the TTY UI:

1. Use the tab key to move to the control panel in the TTY main window (FIGURE 5-1).

2. Select quit and press Return.

The quit menu appears (FIGURE 5-19).

3. To quit the TTY UI, select **Quit UI** and press **Return**.
4. To completely exit SunVTS, select **Quit UI and Kernel**.

FIGURE 5-19 Quit Option

```
quit          reprobe
*****quit*****
Quit UI and Kernel
Quit UI only
*****[DONE:ESC]*****
```


Using the SunVTS Command-Line Interface

This chapter describes the command-line interface used in controlling SunVTS test sessions. Topics include:

- [“Command-Line Interface Overview” on page 99](#)
- [“Command Options” on page 100](#)

Note – The man pages for these commands are installed in the `/usr/sunvts/man/man1m` directory.

Command-Line Interface Overview

Note – SunVTS must first be started before selecting a user interface. See [“Starting SunVTS” on page 14](#).

`vtscmd` is a UNIX shell application that enables you to send a single command to the SunVTS kernel (`vtsk`) from the command line. The SunVTS kernel processes the command and sends the response back to the command line.

To control a SunVTS test session, `vtsk` must first be running. While the test is running, you can control the session by using the `vtscmd` commands.

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. This action returns a reply back in the form of a string of characters.

`vts_cmd` enables you to send commands and receive replies from a UNIX command-line. `vtstalk` is a utility used to establish a connection to the SunVTS kernel.

Command Options

`vts_cmd` and `vtstalk` uses the commands listed in this section. In all cases, the commands (and any command arguments) must follow `vts_cmd` (for `vts_cmd` commands) or `vtstalk(hostname)->` (for `vtstalk` commands). For reference, see the example section in each command option listed.

Note – The examples sections show `vts_cmd` commands. If using `vtstalk` commands, replace `./vts_cmd` with `vtstalk(hostname)->` in the below examples.

`get_agent_summary`

Displays the summary of the agent in the test machine. The information includes the agent machine name, the operating system running on the agent, the version of SunVTS software currently installed on the agent, and the health of the agent.

Example:

```
# ./vts_cmd get_agent_summary
Host Name:diag035,OS:SunOS 5.10,Vts
Version:SunVTS7.0build6,Status:testing/elapsedTime=000.30.56/Error=1
```

`get_host_config`

Displays the configuration information of test machine.

Example:

```
# ./vts_cmd get_host_config
Host Machine Name:diag035;Operating System:SunOS 5.10;Machine
Type:sun4u;IP Address:10.12.150.35;Number of
Processors:1;Architecture:sparc;Hardware Platform:SUNW,Sun-Blade-1500
```

`get_status` *test-name*

Displays the testing status information of the system. If you specify the name of a test, `get_status` displays the status information of the test.

Example:

To get the global status of the system, type:

```
# ./vts_cmd get_status
idle/elapsedTime=000.00.00/Error=0
```

To get the status of a test, append the test name to `get_status`.

```
# ./vts_cmd get_status Memory
Memory:idle/pass=0/error=0/elapsedTime=000:00:00
```

`list_tests`

Displays all the tests that have been successfully probed by the system. The display information further shows all the related details of the test. The details include information on whether the test is selected or not, its scheduling policy (time or pass based), the stress factor (high or low), the progress percentage of the test (when executed), and its status (idle, testing, completed, or failed).

Example:

```
# ./vts_cmd list_tests
Test:Processor;Enabled:False;Scheduling Policy:Time=
600;Stress:high;Progress:100;
Status:completed(Pass=0/Error=0);
Test:Memory;Enabled:True;Scheduling Policy:Pass=
2;Stress:low;Progress:50;
Status:testing(Pass=1/Error=0)
```

`enable_test` *test-name* /*all*

Enables a test on the listed test name or all the tests if the word `all` is selected next to the `enable_test` command. When a test enabled, `vts_cmd` displays the word `DONE`. If the command fails, an error string is displayed along with the usage help.

Example:

```
# ./vts_cmd enable_test Processor  
DONE
```

`disable_test test-name /all`

Disables a test or all the tests. When a test disabled, `vts_cmd` displays the word `DONE`. If the command fails, an error string is displayed along with the usage help.

Example:

```
# ./vts_cmd disable_test Processor  
DONE
```

`get_mode`

Displays the current test mode - Online Stress or System Exerciser or Component Stress.

Example:

```
# ./vts_cmd get_mode  
Test Mode: System Exerciser
```

`set_mode [mode]`

To change test mode, provide the mode name (Online Stress or System Exerciser or Component Stress) as per the testing requirement. If SunVTS successfully sets the test mode, `vts_cmd` displays the word `DONE`.

Example:

```
# ./vts_cmd set_mode [Online Stress]  
DONE
```

If setting the options fails (for example, an error due to format or a misspelled word), an error string displays with the usage help.

Example:

```
# ./vts_cmd set_mode [Online Stress]
Usage: 'vts_cmd set_mode [modeValue] where 'modeValue' can be Online
Stress/System Exerciser/Component Stress'
```

get_global_options

Displays the global options that apply to all the tests. The parameters for the global options includes duration of testing (in minutes), and verbose option (whether enabled or disabled). The parameters are defined as type of display, the chosen or default value, and the choices or range of the values.

Example:

```
# ./vts_cmd get_global_options
Duration of Testing, NUMERIC, 20, 0/99999;
Verbose, CHECKBOX, Enable
```

set_global_options *[Duration of Testing:<0-99999 mins>,Verbose:Enable/Disable]*

To set the global options, you must provide all the option parameters (duration of testing and verbose) with a value in the format shown in the syntax. You can determine the values from the options listed by the `get_global_options` command, as per testing requirements. If SunVTS successfully sets the global options, `vts_cmd` displays the word `DONE`. If setting the options fails, an error string displays along with the usage help.

Example:

```
# ./vts_cmd set_global_options [Duration of
Testing:60,Verbose:Enable]
DONE
```

Setting of global options can fail if there is a misspelled parameter or the value of a parameter is missing. In the following example, the parameter `Verbose` is typed as `Verb`, so the option was not set.

```
# ./vts_cmd set_global_options [Duration of Testing:60,Verb:Enable]
Format Error: Verb is not a valid value.
Usage: 'vts_cmd set_global_options [Duration of Testing:<0-
99999>,Verbose:Enable/Disable]'
```

Setting global options is not allowed when the vts kernel is not in the idle state. An error will be displayed as shown.

```
# ./vts_cmd set_global_options [Test Mode:System Exerciser, Duration
of Testing:60,Verbose:Enable]
ERROR: Setting of options allowed only when vts kernel is idle.
```

`get_test_options` *test-name*

Displays the test options. The parameters for the test options includes stress factor (high or low), scheduling policy (test passes or test time in minutes), and error limit. The parameters have been defined under type of display, the chosen or default value, and the choices or range of the values.

Example:

```
# ./vts_cmd get_test_options Processor
Stress, RADIO, low, low/high;
Scheduling Policy, RADIO, Test Time,
[Test Time, NUMERIC, 10, 0/99999]
Test Passes, NUMERIC, 0, 0/99999];
Error Limit, NUMERIC, 1, 0/99999
```

`set_test_options` *test-name* [*Stress:low/high,Scheduling Policy:Test Time<mins>/Test Passes,Test Time<mins>:x*,Test Passes:y*,Error Limit:z**]

Where x,y, and z denote time, pass, and error count, respectively.

To set the test options, you must provide all the option parameters (stress, scheduling policy and error limit) with a value in the format shown. You can determine the values from the options listed by the `get_test_options` command, as per testing requirements. On success with setting the test options, `vts_cmd` displays the word `DONE`. If setting the options fails, an error string displays along with the usage help.

Example:

```
# ./vts_cmd set_test_options Disk [Stress:high,Scheduling Policy:Test
Time,Test Time:50,Test Passes:0,Error Limit:2]
DONE
```

Setting options can fail if a parameter is misspelled, or the value of a parameter is missing. In the following example, the error limit parameter is missing. For this reason, the setting of that test option failed.

```
# ./vts_cmd set_test_options Disk [Stress:high,Scheduling Policy:Test
Time,Test Time:50,Test Passes:0,Error Limit:]
Format Error: blank space is not a valid value
Usage: 'vts_cmd set_test_options TestName[Stress:low/high,Scheduling
Policy:Test Time<mins>/Test Passes,Test Time<mins>:x,Test Passes:y,Error
Limit:z]',
        where x,y and z denotes time, pass and error count
```

Setting global options is not allowed when the vts kernel is not in the idle state. An error displays as shown.

```
# ./vts_cmd set_test_options Disk [Stress:high,Scheduling Policy:Test
Time,Test Time:50,Test Passes:0,Error Limit:2]
ERROR: Setting of options allowed only when vts kernel is idle.
```

`get_device_options [testname.devicename]`

Displays the devices and the options exported for the test.

Example:

```
# ./vts_cmd get_device_options Disk.c3d0p
Iostate, RADIO, Disable, Enable/Disable;
Write Threshold(KB/s), TEXT, 0, 0;
Read Threshold(KB/s), TEXT, 0, 0
```

`set_device_options [testname.devicename: Exported Variable1:Value1, Exported Variable2:Value 2,...]`

To set the device options exported for the test that you need to provide the command with testname and its device name with the exported variable and its value.

Example:

```
# ./vts_cmd set_device_options Disk.c3d0p [Iostat:Enable]
DONE

# ./vts_cmd set_device_options Disk.c3d0p0 [Iostat:Enable, Write
Threshold\ (KB/s\):30000]
DONE
```

`save_session [overwrite:yes/no, defaultMode:test mode, name:session name]`

To create a new session file you must provide the name of the session as an input parameter. The command has two more attributes: `overwrite:yes/no` and `defaultMode:test mode`. A value of `yes` for the option `overwrite` will overwrite the existing session file (default value is `no`). You can use any of the test modes (Online Stress, System Exerciser, or Component Stress) as a value for `defaultMode`. Whatever the default mode, the session file saves all the options separately for each of the three modes. The default mode only decides which mode will be active once the session is loaded. If you do not provide any input for the `defaultMode` option, then SunVTS will make the current selected Test Mode as the default active Test Mode. On success, command returns `DONE`.

Example:

```
# ./vts_cmd save_session [overwrite:yes,defaultMode:Online Stress,
name:sunvts_session]
DONE
```

In case of misspelled parameters or a missing value, an appropriate usage error message is returned by the command.

Example:

```
# ./vts_cmd save_session [overite:yes,defaultMode:Online Stress,
name:sunvts_session]
Format Error:overite is not a valid value.
```

Usage: `vts_cmd save_session [overwrite:yes,defaultMode:Component Stress, name:vijay]` where the *name* argument is the required parameter.

`list_sessions`

The `list_sessions` can be used to see the list of available session files in the system, and then you can load any session file based on the availability. The command will return the “,” separated list of all the existing session files available in system.

Example:

```
# ./vts_cmd list_sessions
sunvts_session1,sunvts1,sunvts3, session_a, session_b
```

`load_session[defaultMode:testMode, name:sessionName]`

To load an existing session file, the `load_session` command needs to be used to load an existing session file. The command has two input options: `defaultMode` and `name`. You must provide the name of the session file for this command to load the session file. The `defaultMode` option is not mandatory, it provides a way to set the `defaultMode` as the active mode and load the session options corresponding to that test mode. With no default mode, SunVTS will load the named session file and make the default test mode stored in the session file as the current test mode for the loaded session file. On success, command will return `DONE`, otherwise an appropriate error will be displayed by the command.

Example:

```
# ./vts_cmd load_session [name:sunvts_session]
DONE
```

In case of misspelled parameters or a missing value, an appropriate usage error message will be returned by the command.

Example:

```
# ./vts_cmd load_session
Format Error: Command is not in a correct format.
```

Usage: `vts_cmd load_session [defaultMode:Component Stress, name:sunVTS]` where the *name* argument is the required parameter.

get_session_name

You can use the `get_session_name` command to get the currently loaded session name. If no session is currently loaded, then command will return a blank string.

Example:

```
# ./vts_cmd get_session_name
sunvts_session
```

delete_session *sessionName*

This command can be used to delete the existing session file. To remove a session file, you must enter the session file name as a parameter to `delete_session`. If the session file does not exist then an error will be thrown. On success, command will return `DONE` and the session file will be deleted.

Example:

```
# ./vts_cmd delete_session sunvts_session  
DONE
```

In case the file does not exist, an error message is displayed.

```
# ./vts_cmd delete_session sunvts_session xyz  
Error: Session xyz does not exist. Please give valid session to remove.
```

reset_session

Once the complete execution of a session file is over, you can use the `reset_session` command to restore the default values of the SunVTS session without quitting from the SunVTS application.

Example:

```
# ./vts_cmd reset_session  
DONE
```

start

Starts all enabled SunVTS tests. A reply with the word `DONE` confirms that the tests have started.

```
# ./vts_cmd start  
DONE
```

stop

Stops all running SunVTS tests. A reply with the word `DONE` means the tests have stopped testing.

```
# ./vts_cmd stop  
DONE
```

reset

Resets all the tests pass, error, and time limit counts to zero. A reply with the word DONE confirms that the values of the parameters have been set to zero.

```
# ./vts_cmd reset  
DONE
```

Reset is only possible if the vtsk system status is idle. If the system status is not idle, then an error is returned.

```
# ./vts_cmd start  
ERROR: Reset allowed only when kernel is idle.
```

reprobe

Probes all the devices on the test machine and updates the SunVTS kernel device list. A reply with the word DONE reply confirms that the vtsk has been informed to reprobe the devices.

```
# ./vts_cmd reprobe  
DONE
```

Reprobing is only possible if the vtsk system status is idle. If the system status is not IDLE, then an error is printed on the screen.

```
# ./vts_cmd reprobe  
ERROR: Reprobe allowed only when kernel is idle.
```

get_modesequencer_options

Displays the Auto mode sequencer options and its values. Type the following command to get the modesequencer options:

```
# ./vts_cmd get_modesequencer_options
Sequencer,          RADIO,      Disable,      Enable/Disable,,
Enabling the Sequencer will switch on the mode sequencer feature;

First,              RADIO,      None,         None/Online Stress/System
Exerciser/Component Stress,, Select the first mode to run in the
sequence;

Second,             RADIO,      None,         None/Online Stress/System
Exerciser/Component Stress,, Select the second mode to run in the
sequence;

Third,             RADIO,      None,         None/Online Stress/System
Exerciser/Component Stress,, Select the third mode to run in the
sequence;

Loop, NUMERIC, 1,    1/99999,,      Specify the loops that the
sequence should keep repeating
```

set_modesequencer_options

Displays the Auto mode sequencer options and its values.

To set the auto mode sequencer options, you must provide all the option parameters with a value as shown in the syntax. You can determine the values from the options listed by the get_modesequencer_options command. If SunVTS successfully sets the modesequencer options, vts_cmd returns the word DONE. If setting the options fails, an error string displays along with the usage help.

```
# ./vts_cmd set_modesequencer_options
[Sequencer:Enable,First:Online Stress,Second:System
Exerciser,Third:Component Stress,Loop:1]
DONE
```

Where each of the modes - Online Stress, System Exerciser and Component Stress can be also typed as O, S and C (case does not matter).

Setting mode sequencer options can fail if one of the options is given a wrong or misspelled value, or if the value is missing. In the following example, a wrong value has been typed in for the option "Sequencer". The command then returns with an error.

```
# bash-3.00# ./vts_cmd set_modesequencer options  
[Sequencer:Ena,First:Online Stress,Second:System  
Exerciser,Third:Component Stress,Loop:1]
```

Format Error: Ena is not a valid value.

Usage: 'vts_cmd set_modesequencer_options

[First:Online_Stress/System_Exerciser/Component_Stress,

Second:Online_Stress/System_Exerciser/Component_Stress,

Third:Online_Stress/System_Exerciser/Component_Stress, Loop:<1-99999>]'

quit

Terminates the SunVTS kernel (vtsk). A reply with the word DONE confirms that the vtsk quit.

```
# ./vts_cmd quit  
DONE
```

Quitting the vtsk is only possible if the vtsk system status is idle. If the system status is not idle, then an error is returned.

```
# ./vts_cmd start  
ERROR: Quit allowed only when kernel is idle.
```


SunVTS 7.0 Tests

Oracle's Sun Validation and Test Suite (SunVTS) software performs multiple diagnostic hardware tests from a single user interface. SunVTS verifies the connectivity and functionality of controllers and devices.

SunVTS 7.0 contains individual tests that support testing of a wide range of products and peripherals. This chapter describes SunVTS 7.0 tests. Topics include:

- [“Test Requirements” on page 113](#)
- [“Disk Test” on page 114](#)
- [“Environment Test” on page 117](#)
- [“Graphics Test” on page 119](#)
- [“Host Bus Adapters Test” on page 120](#)
- [“Interconnect Test” on page 121](#)
- [“IOports Test” on page 122](#)
- [“Media Test” on page 123](#)
- [“Memory Test” on page 124](#)
- [“Network Test” on page 125](#)
- [“Processor Test” on page 129](#)
- [“Removable Disk Test” on page 115](#)

Test Requirements

The SunVTS 7.0 tests are supported on Solaris 10 5/08 and later releases. The operating system kernel must be configured to support all peripherals that are to be tested.

Some tests have special requirements, such as connectors, installation of test media, or the availability of disk space. These requirements are listed for all the applicable tests.

Disk Test

Description

The Disk test verifies the functionality of different types of disk drives such as ATA, SATA, SAS, and SSD using Media subtest and File System subtest.

Disk Subtests

■ Media subtest

The Media subtest selects the biggest partition on the disk to run these tests: `ReadOnly`, `CompareRead`, or `WriteRead`. If the disk partition has a file system on it, disktest will run the `ReadOnly` or `CompareRead` test. If the disk is raw, then Media subtest might run `WriteRead` if the unsafe testing is enabled. Each of the three modes can run two different methods of disk testing: synchronous I/O and asynchronous I/O.

Disk test runs the `ReadOnly` test by default. To run the `Write/Read` test on the disks, you must enable the unsafe testing option. The Unsafe option should be used with care. Unsafe testing will overwrite existing data on the hard disks. If the unsafe testing is enabled, the test automatically selects the largest partition for running the `Write/Read` test. The selection is based on the following rules:

- The disk should be raw.
- No partition should have a file system on it.
- The disk should not have an SVM or Veritas volume on it.

If any of the above conditions are not met, Disk test runs `ReadOnly` test on the disk partition. For more information about Unsafe test options, see [“Safe and Unsafe Options” on page 51](#).

Note – To run the `Write/Read` test, be sure that you remove the file system on all the partitions of the disk.

■ File system subtest

The file system subtest verifies the disk file system integrity. This subtest exercises mounted disk partitions carrying the file system. By default, the test runs only on system-mounted partitions, it does not premount any additional partitions. If you want SunVTS to premount all of the unmounted partitions that have a file system, you have to set the environment variable `BYPASS_FS_PROBE` to 0 (zero). The test creates two temporary files of the size specified by the File System File Size, writes the data patterns, and compares the two files against each other.

Test Modes

The Disk test supports the following test modes:

Test Mode	Description
Online	Test performs a basic check verifying the functionality of the connected disk devices.
System Exerciser	Test runs with optimal stress on the disks.
Component	Test runs more stressful tests on all disks.

Test Setup

No special setup is required to run this test.

Test Options

The Disk Devices option lists the devices registered under Disk test options. This option provides a way to Select or Deselect the devices.



Removable Disk Test

The Removable Disk test verifies the functionality of USB flash drives, CF (Compact Flash) cards, secure digital (SD) cards and USB disks using subtests like Media, File System, and Asynchronous I/O.

Media subtest

Media subtest runs ReadOnly, ReadCompare, and WriteRead tests on different USB storage devices, CF & SD cards. Each of the three modes can run two different methods of testing: synchronous I/O and asynchronous I/O.

If you enable unsafe testing, the test selects partitions automatically and runs WriteRead tests. The Unsafe option should be used with care. Unsafe testing can cause data corruption. The test will always run ReadOnly tests by default.

The test will select a suitable raw partition to run the WriteRead tests. Partitions for WriteRead tests are selected based on the following criteria:

The partition will not be selected if it is mounted or has a file system.

For more information about Unsafe test options, see [“Safe and Unsafe Options” on page 51](#).

File System Subtest

The file system subtest verifies the file system integrity. This subtest exercises mounted partitions carrying the file system. By default, the test only runs on system-mounted partitions, it does not premount any additional partitions.

If you want SunVTS to premount all of the unmounted partitions that have a file system, you have to set the environment variable `BYPASS_FS_PROBE` to 0 (zero).

The test creates two temporary files and the file sizes are specified by the File System.File Size, writes the data patterns, and compares the two files against each other.

Test Modes

The Disk test supports the following test modes:

Test Mode	Description
Online Test	Performs a basic check verifying the functionality of the connected devices.
System Exerciser	Test runs with optimal stress on the devices.
Component	Test runs more stressful tests on all devices.

Environment Test

The Environmental test exercises and validates the following system hardware:

- Environmental subsystems, which include system fan, LEDs, front panel, keyswitch, power supply, and temperature sensors. The environmental tests these subsystems on the following products:
 - Oracle's Sun Enterprise 450 product line
 - Sun Enterprise 250 systems
 - Oracle's Sun Blade 1000/2000 systems
 - Sun Blade 100/150 systems
 - Sun Blade 1500/2500 systems
 - A70 workstation systems
 - Oracle's Sun Fire 280R product line
 - Sun Fire V440, V445, V240, V245, V210, V215, V250 systems
 - Oracle's Sun Netra 210, 240, 440 systems
- All x86 systems with Base Management Controller (BMC card).
- Remote System Control (RSC) feature, which is integrated on the Sun Enterprise 250 and on the next-generation RSC 2.0 plug-in card introduced with the Sun Fire 280R line.
- Exercises the Advanced Lights-Out Management (ALOM) feature, which is integrated into the Sun Fire V210, V240, V215, V245, V440, and V445, and Sun Netra 240 and 440 systems.
- I²C bus, proper placement of various I2C devices, their operation, status and data integrity.
- Display functions, parameters of an IEEE 1394 digital camera, such as vid mode, frame rate, and frames received.
- SunPCi II card, which is a x86 PC processor embedded in an add-on card, runs bridge and system diagnostics tests.
- PCMCIA memory card, PKCS#11 token mechanism, sensors of the base management controller (BMC).
- Host-to-system controller interface in Sun high-performance UltraSPARC T1 (chip multithreaded [CMT] multicore processor) CPU-based entry-level servers.
- Alarm card on Sun Netra 210, Sun Netra 240, and Sun Netra 440 servers.
- IPMI bus and its associated components, such as sensors and FRUPROMs in the local I²C bus, and DIMM SPD information.

Test Requirements

You must have the `SUNWpiclh`, `SUNWpiclr`, `SUNWpiclu`, and `SUNWpiclx` `picl` packages installed correctly before running the test to verify I²C devices. Verify that the `picld` daemon is running by typing:

```
# ps -ef | grep picld
root 100077 1 0   Sep 11 ? 23:40 /usr/lib/picl/picld
```

If the daemon is not running, restart it by typing the following:

```
# /etc/init.d/picld start
```

To test the LG PC-10 camera, the system must already be running a window environment, such as JDS. If the system has no windowing environment, or is only displaying the login window, the test will neither pass nor fail. If you are working in JDS, you can change your system to 24-bit depth by editing the file `/usr/dt/config/Xservers` or `/etc/dt/config/Xservers`. The file `/etc/dt/config/Xservers` overrides the file `/usr/dt/config/Xservers`, which currently supports the LG PC-10 camera.

Edit the appropriate file to include the following line (type in one continual line):

```
:0   Local local_uid@console root /usr/openwin/bin/Xsun :0
-nobanner -dev /dev/fbs/ffb0 defdepth 24 defclass TrueColor
```

X Windows for Microsoft Windows must be shut down before testing a SunPCi II card.

Test Modes

The Environment test supports the following test modes:

Test Mode	Description
Online	Test performs the basic checks on the hardware present.
System Exerciser	Test performs read-only and runs some selftests.
Component	Test performs read-write and all other subtests. Full set of tests are run.

Note – The Environment test is not available for all subsystems for all platforms. The coverage is dependent on the availability of system interfaces to the hardware.

Graphics Test

The Graphics test verifies the functionality of graphics adapters. Currently supported adapters include:

- SPARC platforms: Oracle’s Sun XVR-2500, XVR-600, XVR-500, XVR-300, XVR-100, Expert3D, and Oracle’s Sun PGX adapters.

Test Modes

The Graphics test supports the following test modes:

Test Mode	Description
Online	Not supported due to the disruptive nature of the graphics tests.
System Exerciser	Supported on both SPARC and x86 platforms. The test exercises the functionality of the graphics adapter along with other testing in the system.
Component	Supported on both SPARC and x86 Platforms. The test puts higher stress and tests each detected graphics adapter individually.

Test Setup

To run the Graphics test as superuser, you must use the console device or start the desktop with the `-ac` option.

▼ To Disable the Solaris Screen Saver

- You must disable all screen savers before testing any graphics device. To disable the Solaris screen saver, type the following at a UNIX prompt:

```
# xset s off
```

▼ To Turn Display Power Management Off

1. Type the following at a UNIX prompt:

```
# xset -dpms
```

▼ To Change the Screen Resolution

- The display resolution must be 1280x1024. To change resolution, type:

```
# fbconfig -res 1280x1024x76
```

Note – To perform `graphicstest` on kfb frame buffers, the desktop must be running on each kfb device in the system.

To run the Graphics test on an x86 system, you must either start the BUI or the GUI and run the Graphics test at the system console, or use the Java remote console from ILOM. See [“Starting the Browser User Interface” on page 25](#) or [“Starting the Graphical User Interface” on page 57](#).

Host Bus Adapters Test

The Host Bus Adapters (HBA) test verifies the functionality of the HBAs in the system.

Test Modes

The Host Bus Adapters test supports the following test modes:

Test Mode	Description
Online	Test performs a basic check verifying the functionality of the devices connected.
System Exerciser	Test exercises the functionality of the HBA along with other testing in the system.
Component	Test puts higher stress and verifies the operation of the HBAs in the system.

Test Setup

Some HBA tests, such as those for Emulex, JNI Fibre-Channel, and Qlogic HBA, provide external connector testing capability. For these tests, you must connect an external connector before enabling the external subtests.

See [Appendix A](#) for information on connectors.



Interconnect Test

The Interconnect test verifies all interconnects and different ASICs on the system board. If you enable unsafe testing, the interconnect test runs WriteRead tests. Unsafe tests can cause data corruption on the hard disks and hence, should be enabled only if unsafe testing is required to be performed. For more information about Unsafe tests see [“Safe and Unsafe Options” on page 51](#).

Test Modes

The Interconnect test supports the following test modes:

Test Mode	Description
Online	Test runs less stressful tests and performs basic checks on the detected hardware.
System Exerciser	Test runs with optimal stress on the detected hardware.
Component	Test runs more stressful tests in sequence on all detected hardware.

Test Setup

No special setup is required to run this test.

IOports Test

The IOports test verifies the following:

- Hardware and software components of the audio subsystem. All Sun supported audio implementations are tested.
- Functionality of PCMCIA modem card and PCMCIA serial I/O socket card.
- Parallel port devices, which include SBus parallel port and IEEE 1286-complaint parallel port.
- Serial asynchronous interface (SAI) card.
- System on-board serial ports and any multiterminal interface (ALM2) boards (`su[0,1]`, `zs[0,1]`, `zsh[0,1]`, `se[0,1]`, `se_hdlc[0,1]`, `asy[0,1]` and `mcp[0-3]`).
- Functionality of the SBus and PCI bus SunHSI boards.
- Functionality of the hardware of the USB subsystem, which includes audio, keyboard, and printer devices.

Test Modes

The IOports test supports the following test modes:

Test Mode	Description
Online	Test performs a basic check verifying the functionality of the devices connected.
System Exerciser	Test runs more stressful tests in parallel on all detected hardware along with the other tests.

Test Setup

The on-board serial ports support connector testing. The connector test requires the null modem and plug connectors described in [Appendix A](#).

There are a variety of connector loopback paths available. The internal loopback paths do not require an external connector. The availability of the loopback paths depends on the device. The `zs(7D)` device has an internal path for synchronous mode and the `se(7D)` device has an internal path for asynchronous mode. The exact type of loopback connector required depends on the system I/O panel.

The connector for the null modem A-to-B option is a female-to-female plug. The pin configuration is the same as the one described for the section 9-Pin to 9-Pin port-to-port loopback cable (see [“9-Pin to 9-Pin Port-to-Port Cable”](#) on page 156).

The connector for the plug A-to-A option is described in the section 9-Pin female single-port loopback plug (see [“9-Pin Female Single-Port Plug”](#) on page 159).

The Sbus and PCI bus SunHSI boards also support connector testing.

[Appendix A](#) of contains more information on SunLink loopback cables and connectors.

Media Test

The Media test verifies the following hardware:

- Optical media drives (CD-ROM, DVD-ROM, CD-RW, DVD-RW, DVD-RAM)
- Tape drives

Note – CD and DVD are the only supported optical disc tests: blue ray discs are not supported in Media test.

Test Modes

The Media test supports the following test modes:

Test Mode	Description
Online	Performs a basic check that verifies the functionality of the connected devices.
System Exerciser	Runs more stressful tests in parallel on all detected hardware along with the other tests.
Component	Runs more stressful tests exclusively on the specified optical disk drive hardware.

Test Setup

Optical disk drives must have the supported media inserted into the optical disk drive. If the media or disk in the drive is changed, then the drive must reprobe to recognize the new media.

Tape media should be present inside the tape drive for tape drive testing.

The media could be blank or used. If the media has been used, the media test performs a read-only test. If the media is blank, media test performs a read/write test. Media test can also perform a CD/DVD read test and a CD/DVD Read/Write test.

Memory Test

The Memory test is used to test the DRAMs, the memory subsystem, virtual memory, and the swap partitions of disks in system. This test achieves large memory bandwidths and performs a mix of read and write MARCH operations to the memory. This test also uses background patterns to bring out pattern sensitive faults in the DRAMs. A part of the test uses the Solaris `valloc` (page aligned) system call to allocate, write, read, and compare virtual memory. The system call causes heavy

paging activity in the system and simulates a stressful environment for the OS. The test reports hard and soft error correction (ECC) errors, memory read errors and addressing problems.

Swap Space Requirements

The swap space required for the Memory test is 400 KB.

Test Modes

The Memory test supports the following test modes:

Test Mode	Description
Online	Test runs less stressful tests and performs basic checks.
System Exerciser	Test runs with optimal stress on the detected hardware.
Component	Test runs more stressful tests in sequence on all detected hardware.

Test Options

The Memory test does not support any options.



Network Test

The Network test verifies the following:

- Checks all the networking hardware on the system (controllers). The machine under test must be attached to a network with at least one other system on the network.
- Performs the Ethernet connector test (Internal, External). You can select Internal/External loopback from the browser user interface (BUI), or TTY for the Network Test. By default, `Loopback:Auto` is selected. Check for the proper connector cable connection before enabling the External Loopback option (see [Test Setup](#)). If `Loopback:Auto` is selected, the test automatically detects the internal or external loopback type.

Note – When the loopback is set to Auto, the test will make an attempt to select external loopback mode. If the external loopback mode can not be selected, the test will select internal loopback mode for testing. If a connector is faulty or loosely connected, the test will fall back to internal loopback mode and will not run the external loopback mode.

- Checks functionality of the SunATM 155 and SunATM 622 SBus and PCI bus adapters.
- Exercises and verifies the proper operation of the Tavor chip and its associated components, such as DDR memory, flash PROM, and internal IB packet transmit/receive circuitry.

Test Modes

The Network test supports the following test modes:

Test Mode	Description
Online	Test runs a basic check verifying the functionality of the devices connected.
System Exerciser	Test runs with optimal stress on the devices.
Component	Test runs more stressful tests in sequence on all devices.

Test Setup

For a complete functionality test of some of the hardware listed above, you might need a connector as described in [Appendix A](#).

Test Options

The Network Tests option lists the devices registered under network test options and provides a way to select or deselect the devices. This section describes options of the Network Test and Network Loopback options.

Network Tests Options

▼ To Select and Deselect Network Tests

1. Follow the steps described in “Changing Test Specific Options” on page 83 (FIGURE 5-5) to reach the network test options.

The Network options menu is displayed (FIGURE 7-1).

2. Use the arrow keys to reach the Network Test Options menu (FIGURE 7-1) and press Return.

3. Use the arrow keys to highlight the Device Options menu and press Return.

By default, all network devices are selected for testing (FIGURE 7-2). Devices can be deselected if you intend to run the test only on specific devices.

4. Press Return to save the selected devices.

If you do not want to save the changes, press Esc. In either case, the curser will return to the Network test options screen.

5. Select Apply to save the Network test options.

Otherwise, press the Esc key to return to the Network test options screen.

FIGURE 7-1 Network Test Options Menu

```
*****Network*****
[*]Disk      Options  idle(Pass=0/E
[*]Environment Options  idle(Pass=0/E
[*]Ioports   Options  idle(Pass=0/E
[*]Media     Options  idle(Pass=0/E
[*]Memory    Options  idle(Pass=0/E
[*]Network   Options  idle(Pass=0/E
[*]Processor Options  idle(Pass=0/E

Options:
Stress:          high
Scheduling Policy: Test Time
Test Time:[0-99999] 0
Test Passes:[0-99999] <5 >
Specific Pass Level:[0-5] 0
Error Limit:[0-99999] 1
Device Options : Click to view choices

Apply Reset
```

FIGURE 7-2 Device Options Menu

```
*****Device Options*****
[*]e1000g0 Options
[*]e1000g1 Options
[*]e1000g2 Options
[*]e1000g3 Options

*****[DONE:RETURN]*****
```

Target Options

The target option enables you to select the target devices registered under Network Test.

1. To reach the Network Test Options menu, follow the steps in [“Changing Test Specific Options” on page 83](#).

The Network Test options menu is displayed ([FIGURE 7-1](#)).

2. Use the arrow keys to select the Network Test Options menu and press Return.
3. Use the arrow keys to highlight the Device Options menu and press Return ([FIGURE 7-2](#)).
4. Select the target that you need, for example e1000g0, and press Return.
5. Enter the target name or IP address ([FIGURE 7-3](#)).
6. Select Apply, to save the target name.
7. Choose your final selection:
 - Select Apply to save the network test options, or
 - Press the Esc key to go back to the network test options screen.

FIGURE 7-3 Target Options Menu



Loopback Options

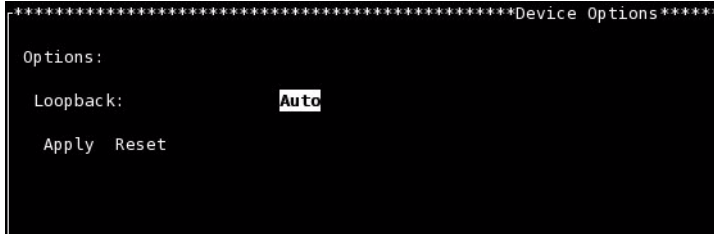
The Loopback option enables you to choose Auto, Internal, and External loopback connections for the devices registered under Network Test.

▼ To Choose Auto, Internal, or External Loopback

1. To reach the Network Test Options menu, follow the steps as described in [“Changing Test Specific Options” on page 83](#) and ([FIGURE 5-5](#)).
2. Use the arrow keys to reach the Network Test Options menu and press Return ([FIGURE 7-1](#)).

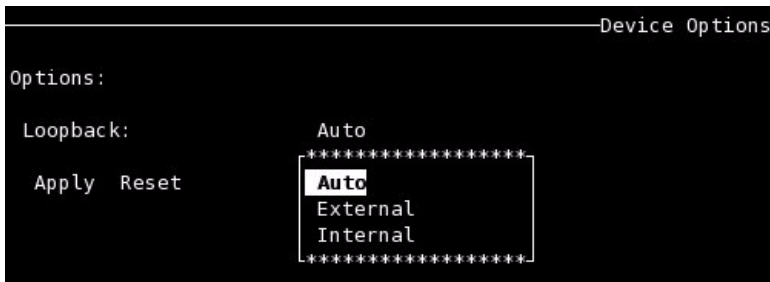
3. Use the arrow keys to highlight the Device Options menu and press Return (FIGURE 7-2).
4. Select a target, for example e1000g1, and press Return.
The Loopback Option menu is displayed (FIGURE 7-2).

FIGURE 7-4 Loopback Option Menu



5. Press Return to select the Internal or External Loopback options (FIGURE 7-5).

FIGURE 7-5 Auto, Internal, or External Loopback Option Menu



6. Select Apply to save the loopback option.
7. Select Apply to save the Network Test Options.
Otherwise, press the Esc key to return to the Network Test Options screen.

Processor Test

The Processor test primarily targets the following features:

- The Integer and Floating-Point Units (IU/FPU)
- The on-chip caches
- Inter-core/ inter-processor communication and cache coherency.

Test Modes

The Processor test supports the following test modes:

Test Mode	Description
Online	Test runs a basic check verifying the functionality of the devices connected.
System Exerciser	Test runs with optimal stress on the devices.
Component	Test runs more stressful tests in sequence on all devices.

SunVTS Bootable CD and USB Environment for Sun x86 Platforms

The bootable SunVTS CD and USB (TTY and BUI) is only supported on Oracle's Sun x86 platforms, based on Intel and AMD processors. This release enables you to use the tool regardless of the native operating systems environment of the host machine, such as Windows or Linux. This chapter covers methods of using the bootable image, some of the limitations that you will encounter in this environment, and recommendations on how to use the bootable CD and USB. See [Chapter 5](#) for information on using the TTY interface used for bootable CD and USB.

This chapter describes the operation of SunVTS through the bootable CD and USB media. Topics include:

- [“Bootable CD and USB Environment Introduction” on page 131](#)
- [“Using Bootable CD or USB Media to Start SunVTS” on page 132](#)
- [“Using the Service Processor to Boot from the SunVTS Image \(Booting from a Virtual Machine\)” on page 134](#)
- [“Limitations” on page 136](#)

Bootable CD and USB Environment Introduction

The bootable SunVTS CD contains an ISO image of the tool which you can use for booting, irrespective of the native environment on the machine under test. An ISO image is an image for an optical disk which has been created using the conventional ISO format. This format is based on the ISO 9660 file system, usually used with CD-ROM media. The format is supported on most platforms. It allows a platform-independent means to describe the contents of a CD.

The bootable SunVTS USB contains the UFS image of the tool, which you can use for booting, irrespective of the native environment on the machine under test. You need to reorder the boot device priority under BIOS so that you can boot from the USB media.

The SunVTS bootable CD and USB image contains the following:

- Solaris operating system released with this tool. Solaris packages not needed by the tool have been removed.
- SunVTS tool for system diagnostics (SUNWvts, SUNWvtsmn, SUNWvtsr, SUNWvtss, and SUNWvtsts)
- HD tool (SUNWhd): Contains the hard disk drive utility for x86 systems which allows you to determine the logical to physical device mapping of the system. For more information on the hd utility, go to <http://docs.sun.com> and search for “hd utility.”
- ssh and bash packages (SUNWssh and SUNWbash)
- Python packages (SUNWPython, SUNWPython-devel, SUNWPython-share, SUNWlxml-python, and SUNWlxml-python)
- IPMI tool package (SUNWipmi)

Using Bootable CD or USB Media to Start SunVTS

You must have a bootable CD either delivered as a part of your system or that you have created a bootable CD. The image can be downloaded from the Sun download center. To burn the ISO image on to the CD, you can use any CD burning device and software.

▼ To Boot Using the CD Using TTY, BUI, or GUI

1. **Insert the disk into the CD drive of the computer.**
2. **Power on the computer.**
3. **Set BIOS to boot from the CD.**

In order to allow the CD-ROM to boot, you have to check that your BIOS is configured properly. In the boot order priorities, the CD drive must be shown before your hard disk.

4. Save the BIOS changes and then exit the BIOS settings.

The machine will boot from the CD and you should then see the SunVTS option to launch the user interface in terminal (TTY) mode or BUI mode.

5. Use the SunVTS TTY, BUI, or GUI interface for testing.

If using the TTY interface for testing, refer to [Chapter 5](#) for information on using the TTY interface.

If using the BUI for testing, do the following:

a. Accept the certificate prompts.

b. At the lockhart login, enter user name `root` and password `solaris`.

c. After login, select the `sunvts` tab that is displayed.

d. Add the host.

Hostname is `localhost` and the password is `solaris`.

e. Start the test.

6. When testing is complete, use the SunVTS TTY or BUI interface to Quit.

You are then returned to the shell prompt.

▼ To Boot Using the USB

1. Insert the USB media to USB port on a computer system, running Solaris x86.

2. Make a temporary directory and get the `vtscopy` and USB Boot image on this directory.

3. Run the `vtscopy` script.

```
# ./vtscopy USB-image-path
```

Follow the steps indicated by the `vtscopy` script. Once the image copy is completed, the script will exit to the prompt.

4. Power on the computer.

5. Set BIOS to boot off the USB Flash disk.

In order to allow the USB to boot, you have to check that your BIOS is configured properly. In the boot order priorities, the USB drive must be shown on top of the HDD list.

6. Save the BIOS changes and exit the BIOS settings.

The machine will boot from the USB, the TTY and BUI option is then displayed.

7. Use the SunVTS TTY or BUI interface for testing.

If using the TTY interface for testing, refer to [Chapter 5](#) for information on using the TTY interface.

If using the BUI for testing, do the following:

a. Accept the certificate prompts.

b. At the lockhart login, enter user name `root` and password `solaris`.

c. After login, select the `sunvts` tab that is displayed.

d. Add the host.

Hostname is `localhost` and the password is `solaris`.

e. Start the test.

8. When testing is complete, use the SunVTS TTY or BUI interface to Quit.

Using the Service Processor to Boot from the SunVTS Image (Booting from a Virtual Machine)

You can use the bootable CD and USB image directly instead of using the physical CD. In Sun systems, the Service Processor (SP) can be used to boot from the bootable CD image. Declare the ISO image as the virtual CD-ROM drive. In the virtual BIOS, you must ask the computer to boot from the CD-ROM. This does not require the actual CD-ROM media for booting. For this method, the booting device can be on the network, in which case, you must specify the IP address.

In the directions below, the Service Processor (SP) on Sun systems is being used as the network device. You should know the SP IP address.

▼ To Do the Boot Process

1. Point your browser to the SP IP address of the machine.

`http://SP-IP address`

2. Log in as root using your root password.

3. Accept the certificates prompts.
4. Click on Remote Control.
5. Click on Launch Redirection (8-bit).
6. Log in again as root using a root password for the SP.
7. Once `connected` appears in the left bottom, enable the keyboard and mouse in Devices at the top of the window.
8. On the redirected ILOM remote console, click Devices.
9. Click on the CD-ROM image (not the CD-ROM).
10. Attach the ISO image (that is, the file with the `.iso` extension) from your path.
11. Reboot the Host through the SP.
`http://SP-IP address`
 - a. Log in as root using your root password.
 - b. Select OK, Run, Accept Certificate, and so on, when prompted.
 - c. Select Remote Control.
 - d. Select Remote Power Control.
 - e. Select Reset.
12. Check the redirected ILOM remote console for shutdown and then reboot messages.
13. Go to the BIOS setup options.
On a Linux system, this is generally done by pressing the F2 key.
14. Use the arrow keys to navigate to Boot Priority.
15. In the Boot Priority menu, make the first boot device the CD media.
16. Save and exit BIOS setup.
On a Linux system, this is generally done by pressing the F10 key.

The boot-loader then looks for the OS binary on the CD-ROM. In SP, however, a virtual CD-ROM has been created by attaching the ISO image as the CD-ROM image. It therefore boots from your ISO image, it may be a bit slow when booting from the ISO image. In case that the redirection on the ILOM remote console expires the session, do a reconnect making sure that the keyboard, mouse, and CD-ROM image in the Devices menu are properly checked.

Limitations

The bootable CD and USB environment presents some challenges for testing of the hardware which are not present in the conventional environment where the tool is run from the disk. In the bootable CD environment, the memory in the system is used to run the image. Below are some limitations that you might encounter with the bootable CD or USB environment.

- Infrastructure Limitations
 - The bootable image only supports the 64-bit environment, that is, the operating system is booted in 64-bit mode.
 - If you want to save the test logs for future reference, the log files need to be saved onto a disk or moved over the network to another location. Two possible setups:
 - Mount the existing disk on the system and copy the files to the disk.
 - Plumb the interface and `ftp` the files to a known system on the network.
- Testing Limitations
 - Memory Test – The amount of physical memory that testing can cover is going to be limited due to the footprint of the bootable image. Additionally, when memory tests are run with other tests in the system, the tests will be allocating and using memory with some reserve amount. This enables all tests to run simultaneously without severe effect on performance.
 - Media Test – Since the system is booted off the CD, testing of that device is done with READ-only operations. Otherwise, the disk might get corrupted.
- Boot USB Limitations
 - The Boot USB image takes a slightly longer time to display compared to the Boot CD image. The progress-bar dots may not be seen in the USB boot image.

▼ To Obtain a Complete FMA Report

- To obtain a complete report of all FMA-generated events, with all error messages, you should use the `VTSreportgenerate` utility. See [“Utilities” on page 171](#).

Connectors

This appendix describes the connectors that aid in testing communication ports.

- [“Connector Overview” on page 138](#)
- [“25-Pin RS-232 Plug” on page 139](#)
- [“25-Pin RS-232 Port-to-Port Cable” on page 141](#)
- [“8-Pin to 8-Pin Cable” on page 143](#)
- [“8-Pin Plug” on page 144](#)
- [“25-Pin Port A-to-Port B Plug” on page 145](#)
- [“25-Pin Port A-to-A Port B-to-B Plug” on page 147](#)
- [“96-Pin Female Connector” on page 149](#)
- [“96-Pin Female Special Connector” on page 151](#)
- [“37-Pin RS-449 Cable” on page 152](#)
- [“37-Pin RS-449 Plug” on page 153](#)
- [“9-Pin Male Single-Port Plug” on page 154](#)
- [“9-Pin Female Single-Port Plug” on page 154](#)
- [“9-Pin to 25-Pin Port-to-Port Cable” on page 155](#)
- [“9-Pin to 9-Pin Port-to-Port Cable” on page 156](#)
- [“NT to TE Cable” on page 156](#)
- [“Twisted-Pair Ethernet \(TPE\) Cable for Fast Ethernet” on page 157](#)
- [“TPE Cable for Gigabit and 10/100 Ethernet” on page 157](#)
- [“9-Pin Male Single-Port Plug” on page 159](#)
- [“9-Pin Female Single-Port Plug” on page 159](#)
- [“9-Pin Male DB-9 External Connector” on page 160](#)
- [“9-Pin Female DB-9 External Connector” on page 161](#)

Connector Overview

Connectors aid in testing communications ports. The connectors take the form of either a single plug or a port-to-port cable with some communications connections shorted (looped-back).

Note – Connectors must be wired properly and connected firmly for the serial port tests to work correctly. Miswired, poorly soldered, or missing connectors can cause erroneous diagnostic error messages.

The following table depicts the pin assignments for most plugs and cables that may be used when testing a system.

TABLE A-1 Connector Pin Assignments

Signal Description	EIA	CCITT #	RS-449 A	RS-449 B	DIN 8 8-pin round	DB9 9-pin	DB25 25-pin	Direction	Alpha ID
Chassis/Frame Ground	AA	101	1	NC*	NC*	NC*	1	None	AA
Transmit Data (TxDa)	BA	103	4	22	3	3	2	Output	BA
Receive Data (RxDa)	BB	104	6	24	5	2	3	Input	BB
Request To Send (RTSa)	CA	105	7	25	6	7	4	Output	CA
Clear To Send (CTSa)	CB	106	9	27	2	8	5	Input	CB
Data Set Ready (DSRa)	CC	107	11	29	NC*	6	6	Input/ output	CC
Signal Ground (SG)	AB	102	19	NC*	4	5	7	None	AB
Data Carrier Detect (DCDa)	CF	109	13	31	7	1	8	Input	CF
Transmit Clock In (TRxCa)	DB	114	5	23	NC*	NC*	15	Input	DB
Receive Clock In (RTxCa)	DD	115	8	26	8	NC*	17	Input	DD
Data Terminal Ready (DTRa)	CD	108	12	30	1	4	20	Output	CD

TABLE A-1 Connector Pin Assignments (*Continued*)

Signal Description	EIA	CCITT #	RS-449 A	RS-449 B	DIN 8 8-pin round	DB9 9-pin	DB25 25-pin	Direction	Alpha ID
External Clock Out (TRxCa)	DA	113	17	35	NC*	NC*	24	Output	DA
Secondary Data Carrier Detect (DCDb)	SCF	122	NC*	NC*	NC*	NC*	12	Input	SCF
Secondary Clear to Send (CTSb)	SCB	121	NC*	NC*	NC*	NC*	13	Input	SCB
Secondary Transmit Data (TxDb)	SBA	118	NC*	NC*	NC*	NC*	14	Output	SBA
Secondary Receive Data (RxDb)	SBB	119	NC*	NC*	NC*	NC*	16	Input	SBB
Secondary Request to Send (RTSb)	SCA	120	NC*	NC*	NC*	NC*	19	Output	SCA

*NC = No connection

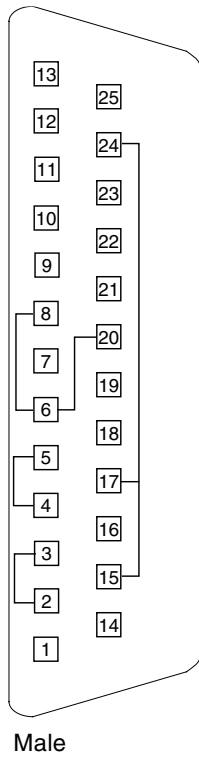
25-Pin RS-232 Plug

FIGURE A-1 shows the RS-232 and RS-423 single-port plug, which is a specially wired male DB-25 connector. Connect the plug to a serial port in the back of the system under test. The following table lists the pin connections for connecting the first plug to the second plug:

TABLE A-2 25-pin RS-232 Plug Wiring Plug Connections

First Plug	Second Plug
Pin 3	Pin 2
Pins 6 and 8	Pin 20
Pins 15 and 17	Pin 24

FIGURE A-1 25-pin RS-232 Plug Wiring Diagram



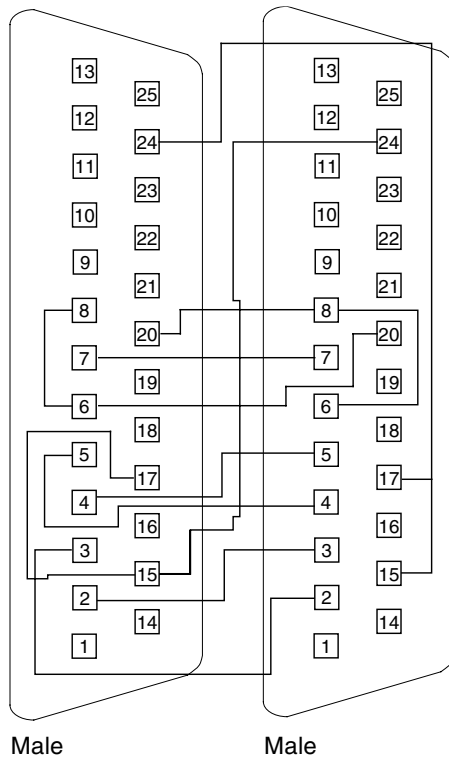
25-Pin RS-232 Port-to-Port Cable

FIGURE A-2 shows how to connect 25-pin RS-232 and RS-423 port to 25-pin RS 232 and RS 423 port cables (two DB-25 connections). Connect the cable to a pair of serial ports in the back of the system under test. Both plugs are male. The following table lists the pin connections for connecting the first plug to the second plug:

TABLE A-3 25-pin RS-232 Port-to-Port Cable Wiring Plug Connections

First Plug	Second Plug
Pin 2	Pin 3
Pin 3	Pin 2
Pin 4	Pin 5
Pin 5	Pin 4
Pins 6 and 8	Pin 20
Pin 7	Pin 7
Pins 15 and 17	Pin 24
Pin 20	Pins 6 and 8
Pin 24	Pins 15 and 17

FIGURE A-2 25-pin RS-232 Port-to-Port Cable Wiring Diagram



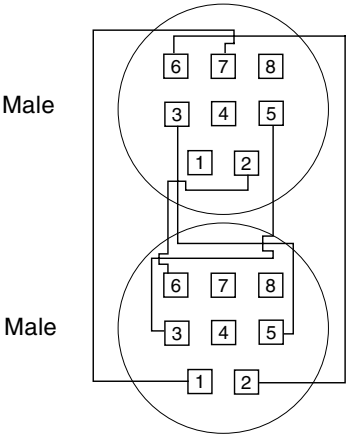
8-Pin to 8-Pin Cable

FIGURE A-3 shows how to connect 8-pin round DIN RS-232 port to RS-423 to 8-pin round-DIN RS-232 and RS-423 port cables. Both plugs are male. The following table lists the pin connections for connecting the first plug to the second plug:

TABLE A-4 8-Pin to 8-Pin Cable Wiring Plug Connections

First Plug	Second Plug
Pin 3	Pin 5
Pin 5	Pin 3
Pin 6	Pin 2
Pin 2	Pin 6
Pin 7	Pin 1

FIGURE A-3 8-Pin to 8-Pin Cable Wiring Diagram



Pin 8, Receive clock In (DD), remains unconnected.

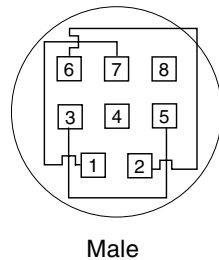
8-Pin Plug

FIGURE A-4 shows how to connect male 8-pin round-DIN RS-232 and RS-423 single-port plugs. The following table lists the pin connections for connecting the first plug to the second plug:

TABLE A-5 8-Pin Plug Wiring Plug Connections

First Plug	Second Plug
Pin 3	Pin 5
Pin 6	Pin 2
Pin 1	Pin 7

FIGURE A-4 8-Pin Plug Wiring Diagram



Pin 8, Receive Clock In (DD), remains unconnected.

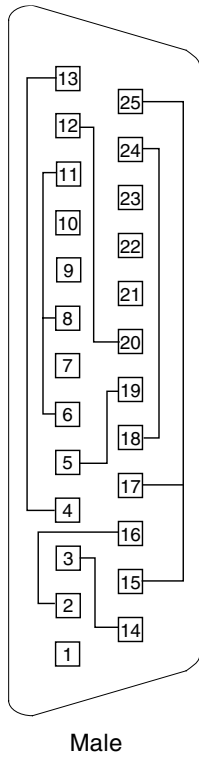
25-Pin Port A-to-Port B Plug

FIGURE A-5 shows how to connect a 25-pin Port A to Port B plug for most systems. The following table lists the pin connections for connecting the first plug to the second plug:

TABLE A-6 Port A-to-Port B Plug Wiring Plug Connections

First Plug	Second Plug
Pin 16	Pin 2
Pin 3	Pin 14
Pin 13	Pin 4
Pin 5	Pin 19
Pins 6 and 8	Pin 11
Pin 12	Pin 20
Pin 18	Pin 24
Pins 15 and 17	Pin 25

FIGURE A-5 Port A-to-Port B Plug Wiring Diagram



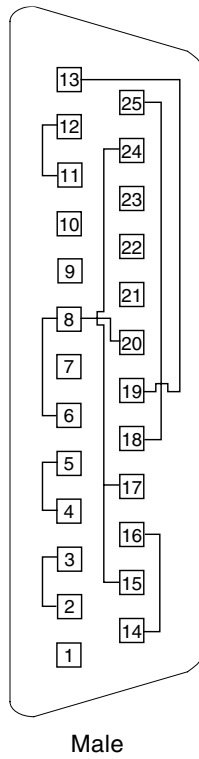
25-Pin Port A-to-A Port B-to-B Plug

If your system has a single communications port to connect it to peripherals, follow the connection instructions in [FIGURE A-6](#) to make a male 25-pin plug for that communications port. The following table lists the pin connections for connecting the first plug to the second plug:

TABLE A-7 Port A-to-A, Port B-to-B Plug Wiring Plug Connections

First Plug	Second Plug
Pin 3	Pin 2
Pin 5	Pin 4
Pins 6 and 8	Pin 20
Pin 12	Pin 11
Pin 13	Pin 19
Pin 16	Pin 14
Pins 15 and 17	Pin 24
Pin 25	Pin 18

FIGURE A-6 Port A-to-A, Port B-to-B Plug Wiring Diagram



96-Pin Female Connector

FIGURE A-7 shows a 96-pin connector that you can order from Sun (part number 370-1366). The following table lists the pin connections for connecting the first plug to the second plug:

TABLE A-8 96-Pin Female Connector Wiring Plug Connections

First Plug	Second Plug
Pins 4 and 12	Pin 77
Pin 5	Pin 20
Pin 6	Pin 36
Pin 10	Pin 11
Pin 13	Pin 16
Pin 14	Pin 15
Pin 18	Pin 19
Pin 21	Pin 24
Pin 28	Pin 60
Pin 29	Pin 68
Pin 30	Pin 34
Pin 37	Pin 40
Pin 38	Pin 39
Pin 42	Pin 43
Pin 45	Pin 48
Pin 46	Pin 47
Pin 52	Pin 78
Pin 53	Pin 55
Pin 54	Pin 75
Pin 58	Pin 59
Pin 61	Pin 64
Pin 62	Pin 63
Pin 66	Pin 67
Pin 69	Pin 72

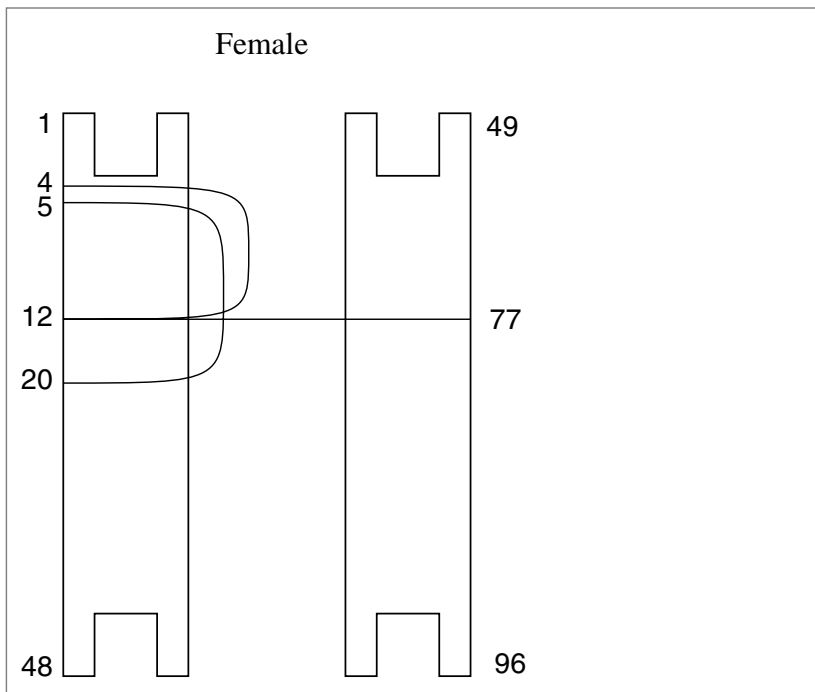
TABLE A-8 96-Pin Female Connector Wiring Plug Connections (*Continued*)

First Plug	Second Plug
Pin 76	Pin 79
Pin 82	Pin 83
Pin 85	Pin 88
Pin 86	Pin 87
Pin 90	Pin 91
Pin 93	Pin 96
Pin 94	Pin 95

The following are the materials used for this plug:

- PCR-E96FA(1)
- PCS-E96LKPA(1)
- 3751 metal plug(1) (9563K42)
- AWG28 Madison cable (8 in. long) UL/CSA Approved

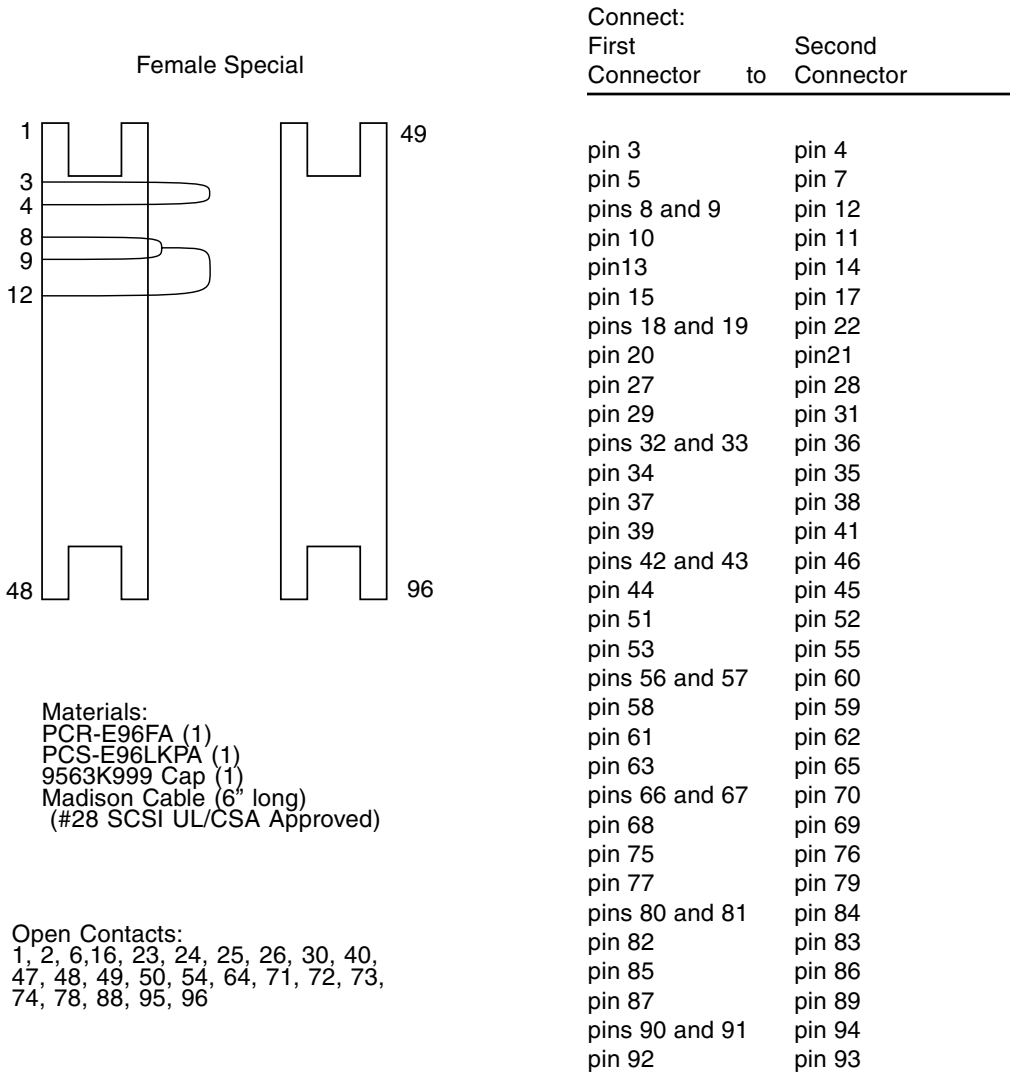
FIGURE A-7 96-Pin Female Connector Wiring Diagram



96-Pin Female Special Connector

You can order this 96-pin connector from Sun (part number 370-1381).

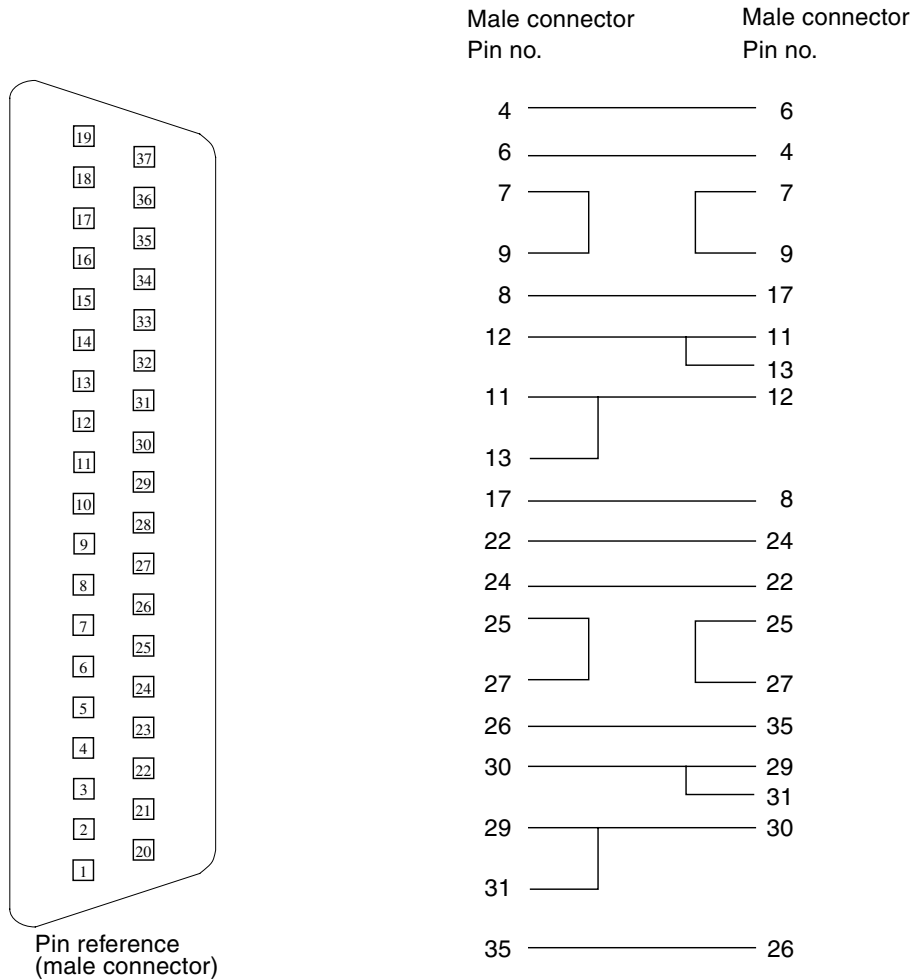
FIGURE A-8 96-Pin Female Special Connector Wiring Diagram



37-Pin RS-449 Cable

Use these wiring instructions for a cable for two 37-pin RS-449 synchronous ports.

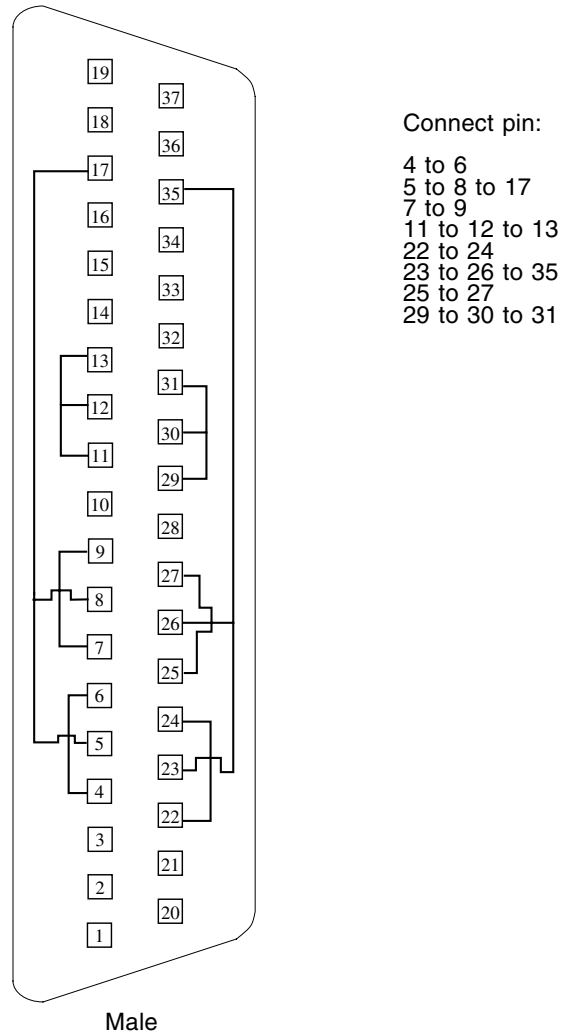
FIGURE A-9 37-Pin RS-449 Cable Wiring Diagram



37-Pin RS-449 Plug

Use these wiring instructions to make a male 37-pin RS-449 plug. This plug is also available from Sun (part number 530-1430).

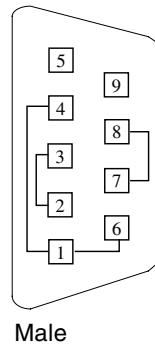
FIGURE A-10 37-Pin RS-449 Plug Wiring Diagram



9-Pin Male Single-Port Plug

Use these wiring instructions for male 9-pin RS-232 and RS-423 single-port plugs.

FIGURE A-11 9-Pin Male Single-Port Plug Wiring Diagram

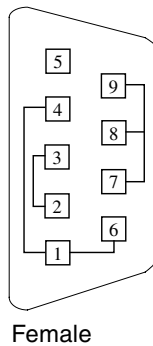


Connect:		
First connector	to	Second connector
Pin 2		Pin 3
Pin 1		Pins 4 and 6
Pin 7		Pin 8

9-Pin Female Single-Port Plug

Use these wiring directions for female 9-pin RS-232 and RS-423 single-port plugs.
Use this plug with the `pcmciaetest`.

FIGURE A-12 9-Pin Female Single-Port Plug Wiring Diagram

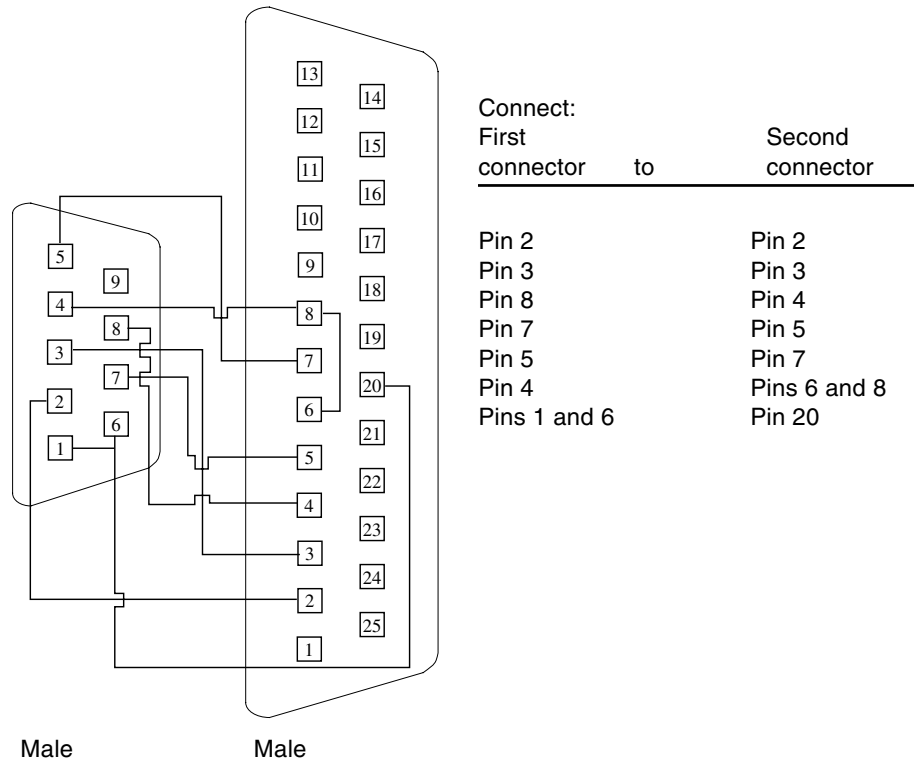


Connect:		
First connector	to	Second connector
Pin 2		Pin 3
Pin 1		Pins 4 and 6
Pin 7		Pins 8 and 9

9-Pin to 25-Pin Port-to-Port Cable

Use these wiring instructions for a 9-pin RS-232 and RS-423 port to 25-pin RS-232 and RS 423 port cables. Both connectors are male.

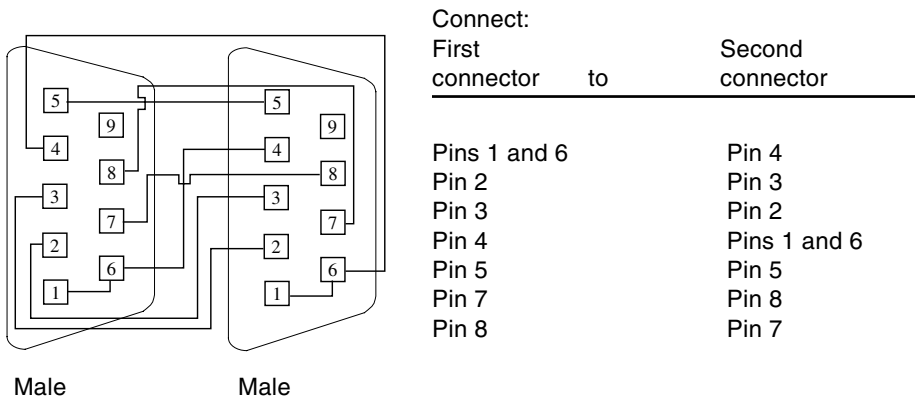
FIGURE A-13 9-Pin to 25-Pin Port-to-Port Cable Wiring Diagram



9-Pin to 9-Pin Port-to-Port Cable

Use these wiring instructions for 9-pin RS-232 and RS 423 port to 9-pin RS-232 and RS-423 port cables. Both plugs are male.

FIGURE A-14 9-Pin to 9-Pin Port-to-Port Cable Wiring Diagram



This cable has no Sun part number assigned to it.

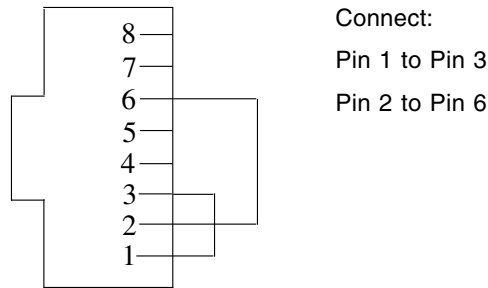
NT to TE Cable

Using two standard RJ-45 connectors, and connect pin 1 to pin 1, pin 2 to pin 2, and so on, for all pins. This is a straight-through connection.

Twisted-Pair Ethernet (TPE) Cable for Fast Ethernet

Use the following wiring instructions for standard RJ-45 connectors for Fast Ethernet. Use cable in `netlbttest` for `eri` devices.

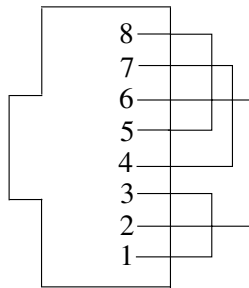
FIGURE A-15 Twisted-Pair Ethernet (TPE) Cable Wiring Diagram for Fast Ethernet



TPE Cable for Gigabit and 10/100 Ethernet

Use the following wiring instructions for RJ-45 plugs for Gigabit and 10/100 Ethernet. This cable is used in `netlbttest` for Gigabit and 10/100 devices.

FIGURE A-16 TPE Cable Wiring Diagram for Gigabit and 10/100 Ethernet



Connect:

Pin1 to Pin 3

Pin 2 to Pin 6

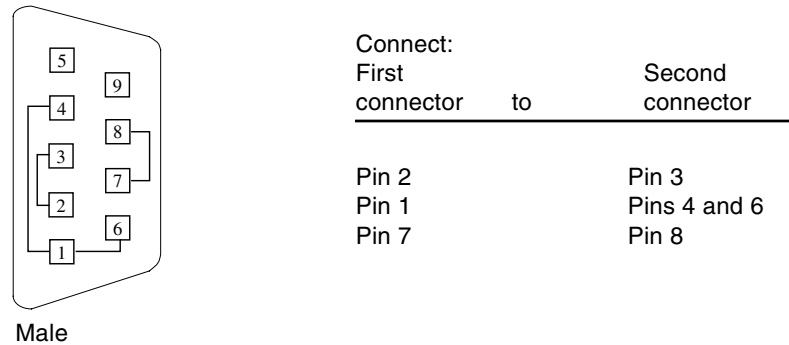
Pin 4 to Pin 7

Pin 5 to Pin 8

9-Pin Male Single-Port Plug

Use these wiring instructions for male 9-pin RS-232 and RS-423 single-port plugs.

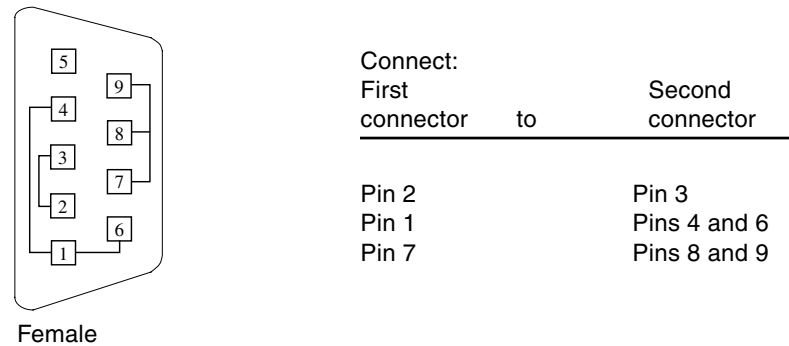
FIGURE A-17 9-Pin Male Single-Port Plug Wiring Diagram



9-Pin Female Single-Port Plug

Use these wiring directions for female 9-pin RS-232 and RS-423 single-port plugs.
Use this plug with the pcmciatest.

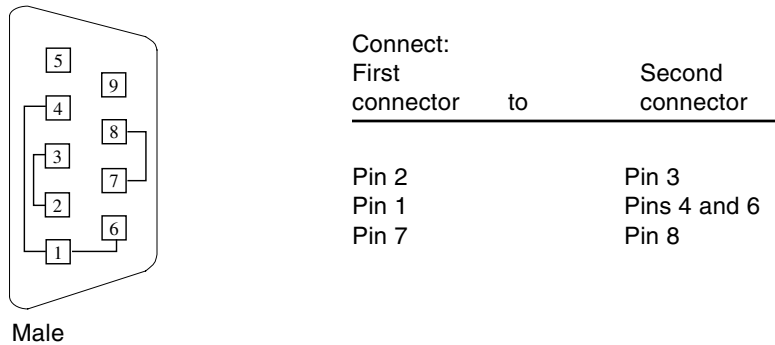
FIGURE A-18 9-Pin Female Single-Port Plug Wiring Diagram



9-Pin Male DB-9 External Connector

Use these wiring instructions for male 9-pin DB-9 external plugs.

FIGURE A-19 9-Pin Male DB-9 External Connector Wiring Diagram



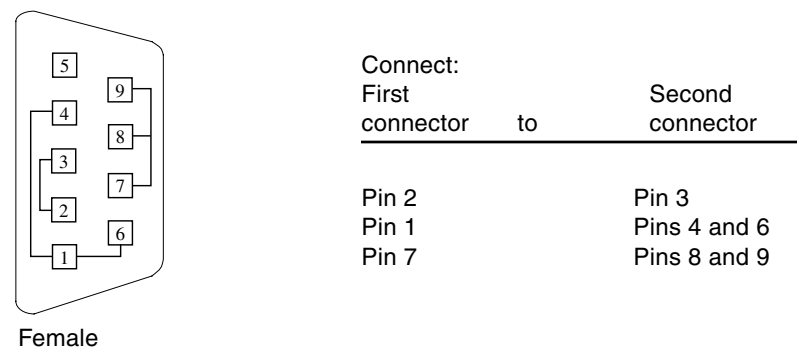
The signals and names for the DB-9 connector are as follows.

Pin	Name	Signal
1	DCD	Data Carrier Detect
2	RxD	Receive Data
3	TxD	Transmit Data
4	DTR	Data Terminal Ready
5	SGND	Signal Ground
6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	RI	Ring Indicator

9-Pin Female DB-9 External Connector

Use these wiring instructions for female 9-pin DB-9 external connectors.

FIGURE A-20 9-Pin Female DB-9 External Connector Wiring Diagram



Frequently Asked Questions (FAQ) and General Recommendations on SunVTS Usage

This appendix provides frequently asked questions regarding SunVTS.

- [“Introduction” on page 163](#)
- [“Frequently Asked Questions \(FAQ\)” on page 164](#)

Introduction

SunVTS is a very powerful and versatile tool which is used by a wide spectrum of users. Although the tool has been designed to perform effectively by default, due to the inherent complexity in hardware testing and wide range of usage models of the tool, there are some guidelines that can help you use the tool even more efficiently. The following questions in this appendix are meant to capture the most frequently asked questions by the tool user community and also to cover some of the typical usage models.

Since the tool is supporting all SPARC and x86 platforms that Sun markets, the range of hardware platforms and configurations is very large. It is not possible to provide inputs and recommendations for each and every platform and configuration separately. Recommendations are made for some typical reference platforms and configurations. You are expected to match the nearest configurations and extrapolate from those configurations.

Frequently Asked Questions (FAQ)

- [“When Should I Use the SunVTS Tool?” on page 164](#)
- [“What Do the Different Modes and the High/Low Within the Modes Mean? When Should I Use Them?” on page 165](#)
- [“How Long Should I Run the Tests?” on page 166](#)
- [“The Test Did Not Fail. Is Everything Ok with My Hardware?” on page 167](#)
- [“Your Tool Caused My System to Panic. Are You Doing Anything that a Normal Application Would Not Do?” on page 168](#)
- [“SunVTS Logs Are Showing a Failure, and the Cause Is Being Attributed to the System Running Out of Resources While Running Tests. Is This a Real Error in My System?” on page 168](#)
- [“I See that the New SunVTS 7.0 Version Is Very Different from the Version I Have Been Using. What Are the Main Differences?” on page 169](#)
- [“I Have Been Using the Previous Generation of SunVTS. I Used to Create Option Files for My Options and Load Them. Can I Use the Same Option Files With the New SunVTS 7.0 Version of the Tool?” on page 169](#)
- [“Who Do I Contact if I Have More Questions?” on page 170](#)
- [“What Is the Meaning of Improved Diagnostics Effectiveness?” on page 170](#)

When Should I Use the SunVTS Tool?

The SunVTS tool is primarily targeted for hardware stress testing. Below are some of the typical uses of the tool.

- Overall System Hardware Validation

The tool can be used as a super application for stressing of the system as a whole. You should run the tool in the “System Exerciser” mode and depending on the stress that you want to put on the system, select “High” or “Low”. The tool will put application level stress on all parts of the system hardware. This could be useful in situations where you are trying to see the system stability, while all the components are stressed at the same time. Some of the situations could be:

- Overall System Stability

SunVTS can be good tool to check the stability of the system, when all parts of the system are under heavy loads. This setup can be further used in four corner testing (testing with margined environmental conditions and mechanical vibrations).

- Validation of New System before Installation

Before putting a new system online or into production, SunVTS can be run on the system to provide functional validation.

- Emission Measurements

For measuring the emission output from a system, you need to make sure that all the parts of the system are running on full cycle. SunVTS can be an effective tool in that situation.

- Environmental Margin Testing of Systems

When we do environmental margin testing, it is necessary to keep all the parts of the system at full speed and then see if there are any fallouts.

- Manufacturing Floor Testing

The tool fits testing needs on the system manufacturing floors. The tool can be completely automated and can be used all the way from board functional testing to system functional testing.

- Validation of a New Component

SunVTS allows tests to be run on the whole system or targeted for a specific component. After repair of a component or installation of a component, the tool can be run in the “Component Stress” mode to specifically target the functionality of the new device. In this mode the tests are more targeted towards the component under stress and the tool ensures that the tests does not have to share any system resources with the other tests.

What Do the Different Modes and the High/Low Within the Modes Mean? When Should I Use Them?

The tool provides three different modes: Online, System Exerciser, and Component Stress. The three modes are meant to serve three different usage models. Below are the usage models for the three modes:

- Online Mode

This mode is meant to be used when you need to test the system in the presence of other application. Since the tests would be running with other applications, they are less stressful on the system resources and have limited coverage.

- **System Exerciser Mode**

This is the mode that can provide overall system hardware validation (mentioned in the last question). All the selected tests run simultaneously, exercising all parts of the system in parallel. You should quit other applications before starting the tests in this mode, mainly because system resources are highly stressed in this mode.

- **Component Stress Mode**

This mode is meant to perform targeted testing of the chosen component. The selected tests are run sequentially, allowing the tests to have full control of system resources to perform the most stressful testing of the component under focus.

In each of these modes the tests can be run in levels “high” or “low”. These levels are meant to give you flexibility to choose the level of stress and coverage that you want. What stress and coverage is put at these levels will vary from one test to another. Typically, tests in the “low” level would be consuming less resources and will be quicker and in many cases will have less coverage. The low level should be used when the goal is to do a quick sanity check or provide less stressful background activity. For example, if the goal is to do an overall system validation, but you know that the CPU is good and you do not want to test it too much, you can change the “Processor” testing to “low” and the rest of the tests at “high”. Another situation would be if you have less time, but you want to do a quick sanity check of a component, you should then use the “low” level. When the goal is stressful exhaustive testing, choose the “high” level.

How Long Should I Run the Tests?

SunVTS gives you the flexibility to run tests based on “Pass” or “Time”. The “Pass” encapsulates the minimum quantum of testing necessary before it can declare a component to be good. So a general guideline is that you need to run at least one pass of the tests that you are interested in.

The time taken by one pass of a test will vary with several factors like size of the hardware configuration, the speed of the system, the architecture of the system, the load on the system, and so on. Therefore, it is not quite possible to accurately gauge or predict the time that would be required for getting at least one pass.

Generally, the minimum time that you need to run the tests include the following:

- System Exerciser (Low): 1 hour
- System Exerciser (High): 2 hours
- Component Stress (Low): 1 hour
- Component Stress (High): 2 hours

If time is not of very high significance and the goal is to do maximum testing that the tool can provide, the recommendation is to run the tests for at least 5 passes in both System Exerciser and Component Stress modes.

The Test Did Not Fail. Is Everything Ok with My Hardware?

May or may not be. In most cases, the faults in the hardware are detected and managed by the system (the hardware plus operating system) and not by the test. SunVTS and its tests are all user-level applications, running on top of the operating system. Any error or failure that is detected and managed by the system may not be visible to the tests. But the occurrence of such errors or failures are logged by the system. So, you should always check the following log files after a run of the testing session. These log files would give a clear picture of any error or failure that might have happened.

- SunVTS Error File

This file contains the failures that are detected and reported by the SunVTS tests. Wherever possible, the message includes possible causes of the error and recommended actions.

- System Log File

These messages are reported by the syslog daemon (`syslogd`). They are logged in the file `/var/adm/messages`. These messages are not necessarily errors, but do tell you about any mishaps that could have occurred while tests were running.

- Output of `fmdump`

Most of the errors that happen on the system are detected, managed and reported by the system (hardware plus operating system), using Proactive Self Healing technology in Solaris. Since the management of the fault happens underneath SunVTS tests, the tests themselves don't see these errors. Solaris provides a utility called "`fmdump`", which allows you to display the errors and faults that the system detected (see the man pages for `fmdump` for more details). After a testing session, look at the output of the following commands to check the errors and faults that happened during the testing session:

```
fmdump -eV
fmdump
```

Your Tool Caused My System to Panic. Are You Doing Anything that a Normal Application Would Not Do?

As mentioned in [“The Test Did Not Fail. Is Everything Ok with My Hardware?” on page 167](#), SunVTS and its tests all run at the user level, on top of the operating system. They do not use any special test specific kernel driver. So they can be considered as any other user level application, whose job is to test the hardware of the system. If the system is experiencing a panic when SunVTS is run, that would indicate a hardware or system software (operating system plus drivers) problem. SunVTS tests merely stepped on the problem (which is its goal). The actions that led to the panic can potentially be caused by any other application as well. So it should be investigated and root cause analysis should be done. It is highly unlikely to be a SunVTS issue.

SunVTS Logs Are Showing a Failure, and the Cause Is Being Attributed to the System Running Out of Resources While Running Tests. Is This a Real Error in My System?

Most probably not. Since SunVTS tests stress the system hardware components very heavily, they also tend to be heavy user of system resources like CPU, memory, system interconnects, network etc. This is specially true in the “System Exerciser” mode when all the tests are run simultaneously. It is quite possible under certain configurations, some tests have to exit due to the lack of enough resources, that they need for doing effective testing. In this case, there is nothing wrong in the system. The recommendation is to disable some of the other tests and run the test that failed again.

In some rare cases the lack of system resources can indicate a problem in the system. In which case when you run that test again, it will fail consistently.

I See that the New SunVTS 7.0 Version Is Very Different from the Version I Have Been Using. What Are the Main Differences?

SunVTS 7.0 is the new generation of the tool and is significantly different from its previous generation. The new version is significantly easier to use and requires less training. The look and feel of the tool has completely changed. This User's Guide should be helpful. Some major differences include the following:

- New Java-Based User Interface

The Motif based GUI in the previous generation of the SunVTS has been replaced with a new web-based GUI (BUI).

- Enterprise Level Support

This feature is only supported on the BUI. From this version, you can monitor and manage SunVTS sessions on multiple hosts through one BUI.

- Tests

Tests in this new SunVTS version has a different meaning and look. SunVTS 7.0 introduces a new software layer between the physical tests (of the previous version) and the user. The new "Tests" are nothing but a combination of earlier physical tests, running in a pre-determined way. All the complexity of option settings of the physical test are hidden under the software layer of the new Tests. The new Tests are very few in number and have very few high level options. The physical tests are identical between the current and previous generation of the tool. These physical tests can still be run manually from a shell prompt.

I Have Been Using the Previous Generation of SunVTS. I Used to Create Option Files for My Options and Load Them. Can I Use the Same Option Files With the New SunVTS 7.0 Version of the Tool?

No, the older option files are no longer supported in SunVTS 7.0. In the SunVTS 7.0, the option settings are significantly easier and happen at a higher level. You no longer need to decide and set options for each and every physical test. These options are automatically set appropriately by the tool. You only needs to provide higher level options such as how much time, how many passes of the test, what level of testing, and so on. These higher level options can be captured in a file called the "session file" which can be stored and loaded.

What Is the Meaning of Improved Diagnostics Effectiveness?

The goal of SunVTS 7.0 diagnostics testing is to be effective by default. SunVTS 7.0 diagnostics improves the effectiveness of testing by these methods:

- Determines the underlying platform where tests are being run.
- Takes into input higher level user goals like time.
- Selects the right combination of tests with their options.
- Runs this combination of tests.

In SunVTS 7.0, with simple and higher level inputs from you, the tool is able to test the hardware appropriately. You do not need to know what combination of tests with associated options are run.

Who Do I Contact if I Have More Questions?

Send email to sunvts-ext@sun.com with your questions and suggestions.

Utilities

This appendix provides information about SunVTS utilities. Topics include:

- [“VTSreportgenerate Utility” on page 171](#)

VTSreportgenerate Utility

The `vtsreportgenerate` script generates a standard report to `stdout` in a verbose and summary format. Output can also be generated in a report text file.

This script generates a standard SunVTS Test Report that provides actionable details on the findings of SunVTS after a stress test. Diagnosed faults and actionable system messages are provided to the user.

Two types of reports can be generated.

- "SunVTS Test Summary Report" that provides test status and faults detected or surfaced due to the SunVTS Test run.
- A detailed "SunVTS Verbose Test Report" that includes all the contents of the Summary Report. The report also includes data such as: specific tests run, number of passes and fails, informational messages from each test and system level messages etc.

Syntax for VTSreportgenerate

```
vtsreportgenerate [ -s | --summary ] [ -v | --verbose ]  
[ -o | --output file ]
```

-s, --summary	Summary Report
-v, --verbose	Verbose Report
-o, --output	Output filename
-h, --help	This usage message

SunVTS Summary Test Report

This is the default report generated with no options passed to vtsreportgenerate. The SunVTS Summary test report will include the following categories of information.:

```
# vtsreportgenerate [ -s | --summary ]
```

SunVTS Test Report Header

The header information will provide data on the test run itself:

```
SunVTS Summary Test Report  
SunVTS Version 7.0 PS5 Build4  
Start Time:   Fri Jan 16 10:00:54 IST 2009  
End Time:     Fri Jan 16 11:00:54 IST 2009  
Hostname:     diag061.goa7.sun.com
```

The Test Status is "PASS" if the error count is not greater than 0. Else, "FAIL".

Test Status

Each Test status is displayed with a Pass or Fail depending on the outcome of the test that was run.

Test Status	
Disk	PASS [FAIL]
Environment:	PASS [FAIL]
Interconnect:	PASS [FAIL]

Media:	PASS [FAIL]
Memory:	PASS [FAIL]
Network:	PASS [FAIL]
Processor:	PASS [FAIL]

Faults and Suspects

Any faults triggered due to a SunVTS Test Run are displayed in the standard Fault Management Architecture format (FMA, or Solaris Predictive Self Healing).

The existing Solaris fault messages are displayed in the output of FMA and the fields and actions are reported from FMA. An example fault report is shown below.

Solaris Fault Example Using `fmadm(1M)`

Faults Detected			
-----	-----	-----	-----
TIME	CACHE-ID	MSG-ID	SEVERITY
-----	-----	-----	-----
Oct 21 07:00:18	cdfe927d-f0d5-c7a6-c4b2-b71eb7b9ccea	PCIEX-8000-8R	Critical
Fault class : fault.io.pciex.device-invreq			
Affects : dev:///pci@3c,0/pci10de,376@f			
degraded but still in service			
FRU : "MB" (hc:///product-id=Sun-Fire-X4540:chassis-id=0000000000:server-id=localhost/motherboard=0)			
faulty			
Description : The transmitting device sent an invalid request.			
Refer to http://sun.com/msg/PCIEX-8000-8R for more information.			
Response : One or more device instances may be disabled			
Impact : Loss of services provided by the device instances associated with this fault			
Action : Ensure that the latest drivers and patches are installed.			
Otherwise schedule a repair procedure to replace the affected device(s). Use <code>fmddump -v -u <EVENT_ID></code> to identify the devices or contact Sun for support.			

High Priority SunVTS Messages

High priority error messages generated by SunVTS are in their own section.

SunVTS Messages
04/28/09 16:10:50 diag062 SunVTS7.0ps6build5: VTSID 1368 Media.cddvdtest.ERROR
rdsk/c4t0d0s0: "DKIOCGMEDIAINFO failed

syslog(3C) Messages

SunVTS stress tests generate high priority error messages in syslog that would be useful to a user to gain insight into causes of failures. Such messages can also be used for empirical analysis and correlations among service actions and human intuitive diagnosis.

Any syslog(3C) messages of a severity like this:

LOG_ERR, LOG_CRIT, LOG_ALERT and LOG_EMERG

will be reported as part of the SunVTS Summary Test Report. See syslog(3C) for more information on the severity of messages.

Syslog Messages

```
Oct 21 07:00:12 rpcbind: [ID 362760 daemon.alert] no value for
config/enable_tcpwrappers (entity not found). Using default "false"
Oct 21 07:00:12 rpcbind: [ID 362760 daemon.alert] no value for
config/enable_tcpwrappers (entity not found). Using default "false"
Oct 21 07:00:12 rpcbind: [ID 362760 daemon.alert] no value for
config/enable_tcpwrappers (entity not found). Using default "false"
Oct 21 07:00:12 rpcbind: [ID 362760 daemon.alert] no value for
config/verbose_logging (entity not found). Using default "false"
```

SunVTS Verbose Test Report

The verbose SunVTS Test Report provides verbose forms of data from the Summary.:

```
# vtsreportgenerate -v | --verbose
```

The header information will provide data on the test run itself.

```
SunVTS Verbose Test Report
SunVTS Version 7.0 PS5 Build4
Start Time:   Fri Jan 16 10:00:54 IST 2009
End Time:     Fri Jan 16 11:00:54 IST 2009
Hostname:     diag067.goa7.sun.com
```


Test Status

Each Test status will be displayed with a PASS or FAIL depending on the outcome of the test that was run. Additionally, the verbose report will include details on individual tests within the test and the number of "passes" and "errors".

Test Status

```
Disk                                passes: 22 errors: 0
c1t1d0p0(disktest)                 passes: 44 errors: 0
c1t0d0(disktest)                   passes: 31 errors: 0
c1t1d0(disktest)                   passes: 35 errors: 0

Environment                         passes: 0  errors: 2118
BMC(bmcenvironment)                passes: 0  errors: 2118

Interconnect                       passes: 28 errors: 0
bus(bustest)                       passes: 169 errors: 0
c1t1d0p0(disktest)                 passes: 56 errors: 0
c1t0d0(disktest)                   passes: 33 errors: 0
c1t1d0(disktest)                   passes: 51 errors: 0
bge0(nettest)                      passes: 977 errors: 0

Media                              passes: 0  errors: 779
c0t1d0(cddvdtest)                 passes: 0  errors: 779

Memory                             passes: 12 errors: 0
  Pass Level 1
    kmem(vmemtest)                 passes: 1  errors: 0
    kmem(vmemtest).1               passes: 1  errors: 0
    mem(pmemtest)                  passes: 18 errors: 0
    mem(pmemtest).1               passes: 18 errors: 0
    kmem(vmemtest)                 passes: 1  errors: 0
    kmem(vmemtest).1              passes: 1  errors: 0
    mem(pmemtest)                  passes: 19 errors: 0
    mem(pmemtest).1               passes: 19 errors: 0
    kmem(vmemtest)                 passes: 1  errors: 0
  ....
  ....
Network                            passes: 884 errors: 0
bge0(nettest)                      passes: 884 errors: 0
bge1(netlptest)                   passes: 3951 errors: 0

Processor                           passes: 2  errors: 0

  Pass Level 1
    CPU(cputest)                   passes: 610 errors: 0
    FPU(fpustest)                  passes: 16 errors: 0
```

l2sram(l2sramtest)	passes: 1	errors: 0
l1cache(l1dcachetest)	passes: 1	errors: 0
CPU(cputest)	passes: 3	errors: 0
l1cache(l1dcachetest)	passes: 1	errors: 0
FPU(fputest)	passes: 2	errors: 0
Pass Level 2		
system(systest)	passes: 1	errors: 0
system(systest)	passes: 1	errors: 0
l1cache(l1dcachetest)	passes: 2	errors: 0
l2sram(l2sramtest)	passes: 1	errors: 0
FPU(fputest)	passes: 77	errors: 0
CPU(cputest)	passes: 151	errors: 0
FPU(fputest)	passes: 91	errors: 0
l2sram(l2sramtest)	passes: 1	errors: 0

Faults and Suspects

Any faults triggered due to SunVTS Test Run are displayed in the standard FMA format.

The existing Solaris Fault messages are predicated on the current output of FMA and the fields and actions reported from FMA. An example of an FMA fault is shown below.

Fault Example using fmadm(1M)

Faults Detected			
TIME	CACHE-ID	MSG-ID	SEVERITY
Oct 21 07:00:18	cdfe927d-f0d5-c7a6-c4b2-b71eb7b9ccea	PCIEX-8000-8R	Critical
Fault class : fault.io.pciex.device-invreq			
Affects : dev:///pci@3c,0/pci10de,376@f			
degraded but still in service			
FRU : "MB" (hc:///product-id=Sun-Fire-X4540:chassis-id=0000000000:server-id=localhost/motherboard=0)			
faulty			
Description : The transmitting device sent an invalid request.			
Refer to http://sun.com/msg/PCIEX-8000-8R for more information.			
Response : One or more device instances may be disabled			
Impact : Loss of services provided by the device instances associated with this fault			
Action : Ensure that the latest drivers and patches are installed.			

Fault Example using fmadm(1M)

Otherwise schedule a repair procedure to replace the affected device(s). Use `fmdump -v -u <EVENT_ID>` to identify the devices or contact Sun for support.

SunVTS Log Messages

SunVTS generates error messages after a stress test. These also aid root cause analysis of faulty HW. These messages are reported as part of the SunVTS Verbose Test Report:

```
01/06/09 10:19:14 diags197.goa78.sun.com SunVTS6.4: VTSID 8011 disktest.FATAL
sda: "Failed lock mtab semaphore. "semop" system call failure, errmsg: Invalid
argument." Probable_Cause(s): <disktest instances exceeds system semaphore
operation limitation (default system limit for seminfo_semmnu = 30)><System
software error> Recommended_Action(s): <Add the line "set
semsys:seminfo_semmnu=0x100" to your /etc/system file and reboot the machine>
<If the problem persists, call your authorized Sun service provider.>
```

```
01/06/09 10:44:24 diags197.goa78.sun.com SunVTS6.4: VTSID 8011 disktest.FATAL
sdc: "Failed lock mtab semaphore. "semop" system call failure, errmsg: Invalid
argument." Probable_Cause(s): <disktest instances exceeds system semaphore
operation limitation (default system limit for seminfo_semmnu = 30)><System
software error> Recommended_Action(s): <Add the line "set
semsys:seminfo_semmnu=0x100" to your /etc/system file and reboot the machine>
<If the problem persists, call your authorized Sun service provider.>
```

syslog(3C) Messages

SunVTS stress tests generate high priority error messages in syslog that help you gain insight into failure causes. The messages can also be used for empirical analysis and correlations among service actions and human intuitive diagnosis. Any syslog(3C) messages of a severity like this:

LOG_NOTICE, LOG_WARNING, LOG_ERR, LOG_CRIT, LOG_ALERT and
LOG_EMERG

will be reported as part of the SunVTS Verbose Test Report. See syslog(3C) for more information on severity of messages. An example is shown below:

```
Apr 24 18:29:00 diag062 automountd[398]: [ID 476599 daemon.warning] srl-cnwd03-02: no NFS service

Apr 27 18:12:17 diag062 gnome-session[733]: [ID 702911 daemon.warning] WARNING:
Could not connect to ConsoleKit: Could not get owner of name
'org.freedesktop.ConsoleKit': no such name

Apr 27 18:21:25 diag062 gnome-session[2182]: [ID 702911 daemon.warning]
EggSMClient-WARNING: Desktop file '/usr/share/gnome/autostart/desktop-print-
management-applet.desktop' has malformed Icon key 'print-manager.png' (should not
include extension)

Apr 27 18:30:25 diag062 gnome-session[2182]: [ID 702911 daemon.warning] WARNING:
Could not connect to ConsoleKit: Could not get owner of name
'org.freedesktop.ConsoleKit': no such name

Apr 27 18:30:36 diag062 scsi: [ID 107833 kern.warning] WARNING:
/pci@0,0/pci108e,4843@1d,7/storage@3/disk@0,0 (sd2):

Apr 27 18:30:36 diag062 scsi: [ID 107833 kern.warning] WARNING:
/pci@0,0/pci108e,4843@1d,7/storage@3/disk@0,0 (sd2):
```

Index

A

adding a new host, 29, 30

B

bootable CD/DVD, 131

browser user interface (BUI), 25

- adding a new host, 29, 30

- host machine view, 28

- Java web console, 27

- login screen, 26

C

command-line interface, 100

connectors, 137

F

FAQ, frequently asked questions, 163, 171

H

harness (`vtstk`), 2

host machine view, BUI, 28

I

interface, browser user (BUI), 25

interfaces

- command-line interface, 100

L

login screen, 26

Logs screen, 52

Q

questions, FAQ, 163, 171

S

safe and unsafe tests, 51, 71

SunVTS 7.0

- bootable CD/DVD, 131

- interfaces, 25, 77

- overview, 1

- terminology, 2

- tests, 113

T

terminology, 2

Test Group screen, 35

- button functions, 39, 61

- setting test parameters, 38

- table description, 38, 59

Test Options screen, 45

tests, 113

- group screen, 35

TTY user interface, 77

U

unsafe and safe tests, 51, 71

V

`vts_cmd` utility, 100

`vtstk`, 2

`vtstkJB`, 2

`vtstalk` utility, 100

W

web console, Java, 27