



Hitachi HiCommand™ Device Manager Command Line Interface (CLI) Application User's Guide

HiCommand™ Device Manager Version 2.3
--

© 2003 Hitachi Data Systems Corporation, ALL RIGHTS RESERVED

Notice: No part of this publication may be reproduced or transmitted in any form or by any electronic or mechanical means, including photocopying and recording, or stored in a database or retrieval system for any purpose, without the express written permission of Hitachi Data Systems Corporation.

Hitachi Data Systems reserves the right to make changes to this document at any time without notice and assumes no responsibility for its use. Hitachi Data Systems products or services can only be ordered under the terms and conditions of Hitachi Data Systems' applicable agreements, including license agreements. All of the features described in this document may not be currently available. Refer to the most recent product announcement or contact your local Hitachi Data Systems sales office for information on feature and product availability.

This document contains the most current information available at the time of publication. When new and/or revised information becomes available, this entire document will be updated and distributed to all registered users.

Trademarks

Hitachi Data Systems and the Hitachi Data Systems design mark are registered trademarks service marks of Hitachi, Ltd.

HiCommand, Hitachi Freedom Storage, and Lightning 9900 are trademarks of Hitachi Data Systems Corporation.

Windows and Windows NT are registered trademarks of Microsoft Corporation.

Java, Solaris, StorEdge, and Sun are trademarks or registered trademarks of Sun Microsystems, Inc.

VERITAS is a registered trademark of VERITAS Software Corporation.

UNIX is a registered trademark of The Open Group in the United States and other countries.

All other brand or product names are or may be registered trademarks, trademarks or service marks of and are used to identify products or services of their respective owners.

Notice of Export Controls

Export of technical data contained in this document may require an export license from the United States government and/or the government of Japan. Contact the Hitachi Data Systems Legal Department for any export compliance questions.

Document Revision Level

Revision	Date	Description
MK-91HC007-P	November 2001	Preliminary Release
MK-91HC007-0	January 2002	Revision 0, supersedes and replaces MK-91HC007-P
MK-91HC007-1P	May 2002	Preliminary Release of Revision 1, supersedes and replaces MK-91HC007-0
MK-91HC007-1	June 2002	Revision 1, supersedes and replaces MK-91HC007-1P
MK-91HC007-2	August 2002	Revision 2, supersedes and replaces MK-91HC007-1
MK-91HC007-3	November 2002	Revision 3, supersedes and replaces MK-91HC007-2
MK-91HC007-4	May 2003	Revision 4, supersedes and replaces MK-91HC007-3

Changes in this Revision

- Added CLI help information and CLI version information (section 2.7 and Table 3.53).
- Revised information on using CLI with HTTPS (section 2.5.1).
- Revised information on CLI case-sensitive instances (throughout document).
- Added CLI memory usage requirements (section 2.8).
- Revised the information on command syntax (section 3.1).
- Added command examples (new Chapter 5).
- Added the following new storage array commands:
 - **ModifyLogicalUnit** command (section 3.2.12).
 - **AddLUSE** command (section 3.2.13).
 - **DeleteLUSE** command (section 3.2.14).
 - **RefreshStorageArrays** command (section 3.2.17).
- Added the following new LUN commands:
 - **AddLunGroup** command (section 3.4.1).
 - **DeleteLunGroup** command (section 3.4.2).
 - **ModifyLunGroup** command (section 3.4.3).
 - **AddWWNForLunGroup** command (section 3.4.4).
 - **DeleteWWNForLunGroup** command (section 3.4.5).
 - **AddWWNGroup** command (section 3.4.11).
 - **ModifyWWNGroup** command (section 3.4.12).
 - **DeleteWWNGroup** command (section 3.4.13).
 - **AddWWNForLun** command (section 3.4.14).
 - **DeleteWWNForLun** command (section 3.4.15).

- Change(s) for the **AddStorageArray** command (section 3.2.1):
 - Updated **userid** and **arraypasswd** parameters.
- Change(s) for the **GetStorageArray** command (section 3.2.2):
 - Added **ReplicationInfo** subtarget.
 - Added **controllersubinfo** parameter for **PortController** subtarget.
 - Modified command such that **LogicalGroup** can be specified in the **pathsubinfo** parameter when parameter subtarget value is **Path**.
- Change(s) for the **DeleteHostStorageDomain** command (section 3.2.5):
 - Added **deletionoption** parameter.
- Change(s) for the **AddLogicalUnit** command (section 3.2.10):
 - Added **lusubinfo** parameter.
 - Changed the **emulation** parameter from “required” to “optional”.
- Change(s) for the **GetLogicalGroup** command (section 3.3.4):
 - Modified **subtarget** parameter to specify **Path**.
- Change(s) for the **AddObjectForLogicalGroup** and **DeleteObjectForLogicalGroup** commands (sections 3.3.5, 3.3.6):
 - Modified **groupelements** parameter to specify **Path**.
- Change(s) for the **AddLun** command (section 3.4.6):
 - Added **wwn** parameter.
 - Updated **domain** parameter.
- Change(s) for the **DeleteLun** command (section 3.4.7):
 - Added **deletionoption** parameter.
- Change(s) for the **ModifyHost** command (section 3.5.3):
 - Updated **wwnlist** parameter.
- Change(s) for the **AddURLLink** command (section 3.6.7):
 - Added **description** parameter.
- Change(s) for the **GetURLLink** command (section 3.6.9):
 - Updated **objectid** parameter.
 - Updated **linkedid** parameter.
- Updated descriptions of **hostmode** and **hostmode2** parameters for the **ModifyPort** command (section 3.2.15).

Referenced Documents

- *HiCommand™ Device Manager Server Installation and Configuration Guide*, MK-91HC002
HiCommand™ Device Manager Server Security Guide, MK-91HC003
HiCommand™ Device Manager Agent™ Installation Guide, MK-91HC019
- *Hitachi Lightning 9900™ V Series User and Reference Guide*, MK-92RD100
Hitachi Lightning 9900™ V Series Remote Console - Storage Navigator User Guide, MK-92RD101
Hitachi Lightning 9900™ V Series LUN Manager User's Guide, MK-92RD105
Hitachi Lightning 9900™ V Series LUN Expansion (LUSE) and Virtual LVI/LUN User Guide, MK-92RD104
- *Hitachi Lightning 9900™ User and Reference Guide*, MK-90RD008
Hitachi Lightning 9900™ Remote Console User Guide, MK-90RD003
Hitachi Lightning 9900™ LUN Manager, LUSE, and SANTinel User's Guide, MK-91RD049
Hitachi Lightning 9900™ Virtual LVI/LUN User's Guide, MK-90RD005
- *Hitachi Thunder 9500™ V Series User and Reference Guide*, MP-92DF601
Hitachi Freedom Storage Thunder 9500 V Series Resource Manager 9500V User's Command Line Interface (CLI), MP-92DF603
Hitachi Freedom Storage Thunder 9500 V Series Resource Manager 9500V User's Guide Graphical User Interface (GUI), MP-92DF605
Hitachi Freedom Storage Thunder 9500 V Series SANTinel 9500V User's Guide, MK-92DF613
- *Hitachi Thunder 9200™ User and Reference Guide*, MK-90DF504
Hitachi Thunder 9200™ Resource Manager 9200 User's Guide, MK-90DF505
Hitachi Thunder 9200™ LUN Security 9200 User's Guide, MK-91DF554
- *Sun™ StorEdge™ T3 Disk Tray Configuration Guide*, 806-4210

Preface

The document describes and provides instructions for using the HiCommand™ Device Manager Command Line Interface (CLI) software on the Hitachi Freedom Storage™ and Sun™ StorEdge™ subsystems. This document assumes that:

- The user has a background in data processing and understands peripheral storage device subsystems and their basic functions,
- The user has read and understands the user guide for the subsystem: *Hitachi Lightning 9900™ V Series User and Reference Guide*, *Hitachi Lightning 9900™ User and Reference Guide*, *Hitachi Thunder 9500™ V Series User and Reference Guide*, *Hitachi Thunder 9200™ User and Reference Guide*, *Sun™ StorEdge™ T3 and T3+ Array Installation, Operation, and Service Manual*.
- The user has read and understands the user guide(s) for the subsystem configuration functions (e.g., *Lightning 9900™ V Series LUN Manager User's Guide*, *Thunder 9200™ Resource Manager 9200 User's Guide*, *StorEdge™ T3 Disk Tray Configuration Guide*).
- The user has read and understands the user guide(s) for the data management functions (e.g., *Hitachi Lightning 9900™ Virtual LVI/LUN User's Guide*).
- The user is familiar with the operating system (OS) that hosts the HiCommand™ Device Manager Command Line Interface (CLI) software (e.g., Solaris™ OS, Windows NT® OS).

Note: The term “9900V” refers to the entire Lightning 9900™ V Series subsystem family (e.g., 9980V, 9970V), unless otherwise noted. The term “9900” refers to the entire Lightning 9900™ subsystem family (e.g., 9960, 9910), unless otherwise noted. The term “9500V” refers to the entire Thunder 9500V subsystem family (e.g., 9570V, 9580V), unless otherwise noted.

Note: The use of HiCommand™ Device Manager and all other Hitachi Data Systems products is governed by the terms of your license agreement(s) with Hitachi Data Systems.

Note: The use of the Sun™ StorEdge™ T3 Array and all other Sun™ products is governed by the terms of your license agreement(s) with Sun Microsystems.

Software Revision Level

This document revision applies to HiCommand™ Device Manager version 2.3.

COMMENTS

Please send us your comments on this document: doc.comments@hds.com.

Make sure to include the document title, number, and revision.
Please refer to specific page(s) and paragraph(s) whenever possible.
(All comments become the property of Hitachi Data Systems Corporation.)

Thank you!

Contents

Chapter 1 HiCommand™ Device Manager Overview

1.1	Overview of HiCommand™ Device Manager	1
1.2	HiCommand™ Device Manager Software Components	3

Chapter 2 Requirements and Specifications

2.1	Requirements for CLI Operations	5
2.2	Storage Subsystem Requirements	5
2.3	Restrictions of CLI Compatibility	6
2.3.1	R2.0 CLI with HiCommand™ Device Manager 2.0	6
2.3.2	R2.0 CLI with HiCommand™ Device Manager 2.1 (and higher)	6
2.4	CLI Command Syntax	8
2.4.1	Launching CLI	8
2.4.2	Format for the Command Line Values	8
2.5	HiCommand™ Device Manager Server URL	10
2.5.1	Using CLI with HTTPS	10
2.6	CLI Return Responses	12
2.7	Displaying CLI Help	13
2.8	CLI Memory Requirements	15

Chapter 3 Specifying Command Parameters

3.1	Description of Syntax	17
3.2	Storage Array Commands	18
3.2.1	AddStorageArray	18
3.2.2	GetStorageArray	19
3.2.3	DeleteStorageArray	21
3.2.4	AddHostStorageDomain	21
3.2.5	DeleteHostStorageDomain	21
3.2.6	AddArrayGroup	22
3.2.7	DeleteArrayGroup	22
3.2.8	AddSpareDrive	23
3.2.9	DeleteSpareDrive	23
3.2.10	AddLogicalUnit	24
3.2.11	DeleteLogicalUnit	24
3.2.12	ModifyLogicalUnit	25
3.2.13	AddLUSE	25
3.2.14	DeleteLUSE	26
3.2.15	ModifyPort	26
3.2.16	ModifyPortController	28
3.2.17	RefreshStorageArrays	28

3.3	Logical Group Commands	29
3.3.1	AddLogicalGroup	29
3.3.2	DeleteLogicalGroup	29
3.3.3	ModifyLogicalGroup	30
3.3.4	GetLogicalGroup	30
3.3.5	AddObjectForLogicalGroup	31
3.3.6	DeleteObjectForLogicalGroup	31
3.3.7	AddLunScan	31
3.4	LUN Commands	32
3.4.1	AddLunGroup	32
3.4.2	DeleteLunGroup	32
3.4.3	ModifyLunGroup	33
3.4.4	AddWWNForLunGroup	33
3.4.5	DeleteWWNForLunGroup	34
3.4.6	AddLun	35
3.4.7	DeleteLun	36
3.4.8	AddWWNForHostStorageDomain	36
3.4.9	DeleteWWNForHostStorageDomain	37
3.4.10	DeleteWWN	37
3.4.11	AddWWNGroup	38
3.4.12	ModifyWWNGroup	38
3.4.13	DeleteWWNGroup	39
3.4.14	AddWWNForLun	39
3.4.15	DeleteWWNForLun	40
3.5	Host Management Commands	41
3.5.1	AddHost	41
3.5.2	DeleteHost	41
3.5.3	ModifyHost	42
3.5.4	GetHost	42
3.5.5	AddHostInfo	43
3.5.6	DeleteHostInfo	43
3.5.7	ModifyHostInfo	44
3.5.8	GetHostInfo	45
3.6	Server Management Commands	46
3.6.1	GetServerInfo	46
3.6.2	GetLogFile	46
3.6.3	ModifyDebugLevel	46
3.6.4	GetDebugLevel	47
3.6.5	GetAlerts	47
3.6.6	DeleteAlerts	47
3.6.7	AddURLLink	48
3.6.8	DeleteURLLink	48
3.6.9	GetURLLink	48

Chapter 4 HiCommand™ Device Manager Properties File

4.1	Using the Properties File to Specify Options	49
4.2	Using the Properties File to Specify Parameters	50
4.3	Changing Specific Properties from the Command Line	51
4.3.1	Diagnostic Logging	51
4.3.2	Message Trace Output	51
4.3.3	Specifying the URL	51
4.3.4	Inputting Requests from XML Documents	52
4.3.5	Using the Message Trace File	52

Chapter 5 Command Examples and Results

5.1	Storage Array Commands	53
5.1.1	AddStorageArray	53
5.1.2	GetStorageArray	54
5.1.3	DeleteStorageArray	71
5.1.4	AddHostStorageDomain	71
5.1.5	DeleteHostStorageDomain	71
5.1.6	AddArrayGroup	72
5.1.7	DeleteArrayGroup	73
5.1.8	AddSpareDrive	73
5.1.9	DeleteSpareDrive	73
5.1.10	AddLogicalUnit	74
5.1.11	DeleteLogicalUnit	76
5.1.12	ModifyLogicalUnit	76
5.1.13	AddLUSE	77
5.1.14	DeleteLUSE	78
5.1.15	ModifyPort	78
5.1.16	ModifyPortController	79
5.1.17	RefreshStorageArrays	79
5.2	Logical Group Commands	80
5.2.1	AddLogicalGroup	80
5.2.2	DeleteLogicalGroup	80
5.2.3	ModifyLogicalGroup	81
5.2.4	GetLogicalGroup	81
5.2.5	AddObjectForLogicalGroup	83
5.2.6	DeleteObjectForLogicalGroup	83
5.2.7	AddLunScan	84
5.3	LUN Commands	85
5.3.1	AddLunGroup	85
5.3.2	DeleteLunGroup	85
5.3.3	ModifyLunGroup	86
5.3.4	AddWWNForLUNGroup	87
5.3.5	DeleteWWNForLUNGroup	87
5.3.6	AddLun	88
5.3.7	DeleteLun	90
5.3.8	AddWWNForHostStorageDomain	91
5.3.9	DeleteWWNForHostStorageDomain	91
5.3.10	DeleteWWN	91
5.3.11	AddWWNGroup	92

5.3.12	ModifyWWNGroup	93
5.3.13	DeleteWWNGroup	93
5.3.14	AddWWNForLun	94
5.3.15	DeleteWWNForLun	94
5.4	Host Management Commands	95
5.4.1	AddHost	95
5.4.2	DeleteHost	95
5.4.3	ModifyHost	95
5.4.4	GetHost	96
5.4.5	AddHostInfo	97
5.4.6	DeleteHostInfo	97
5.4.7	ModifyHostInfo	98
5.4.8	GetHostInfo	99
5.5	Server Management Commands	100
5.5.1	GetServerInfo	100
5.5.2	GetLogFile	100
5.5.3	ModifyDebugLevel	101
5.5.4	GetDebugLevel	101
5.5.5	GetAlerts	101
5.5.6	DeleteAlerts	102
5.5.7	AddURLLink	102
5.5.8	DeleteURLLink	103
5.5.9	GetURLLink	103

Chapter 6 Troubleshooting

6.1	Troubleshooting	105
6.2	Hitachi Data Systems Support Center	106

Acronyms and Abbreviations	107
-----------------------------------	-----

List of Figures

Figure 2.1	Modified java Command	11
Figure 2.2	Newly Created Batch File Command Line	11
Figure 2.3	Example of Basic Help Information for CLI	13-14
Figure 2.4	GetstorageArray in Windows® Systems	15
Figure 3.1	Example of CLI Help for AddLun Command	17

List of Tables

Table 2.1	AddObjectForLogicalGroup Examples	6
Table 2.2	DeleteObjectForLogicalGroup Examples	7
Table 2.3	HiCommand™ Device Manager CLI Variable Syntax	8
Table 2.4	CLI Application Options	9
Table 2.5	Return Variables	12
Table 3.1	AddStorageArray Command Parameters	18
Table 3.2	GetStorageArray Command Parameters	19-20
Table 3.3	DeleteStorageArray Command Parameters	21
Table 3.4	AddHostStorageDomain Command Parameters	21
Table 3.5	DeleteHostStorageDomain Command Parameters	21
Table 3.6	AddArrayGroup Command Parameters	22
Table 3.7	DeleteArrayGroup Command Parameters	22
Table 3.8	AddSpareDrive Command Parameters	23
Table 3.9	DeleteSpareDrive Command Parameters	23
Table 3.10	AddLogicalUnit Command Parameters	24
Table 3.11	DeleteLogicalUnit Command Parameters	24
Table 3.12	ModifyLogicalUnit Command Parameters	25
Table 3.13	AddLUSE Command Parameters	25
Table 3.14	DeleteLUSE Command Parameters	26
Table 3.15	ModifyPort Command Parameters (continues on the next page)	26
Table 3.15	ModifyPort Command Parameters (continued)	27
Table 3.16	ModifyPortController Command Parameters	28
Table 3.17	RefreshStorageArrays Command Parameters	28
Table 3.18	AddLogicalGroup Command Parameters	29
Table 3.19	DeleteLogicalGroup Command Parameters	29
Table 3.20	ModifyLogicalGroup Command Parameters	30
Table 3.21	GetLogicalGroup Command Parameters	30
Table 3.22	AddObjectForLogicalGroup Command Parameters	31
Table 3.23	DeleteObjectForLogicalGroup Command Parameters	31
Table 3.24	AddLunScan Command Parameters	31
Table 3.25	AddLunGroup Command Parameters	32
Table 3.26	DeleteLunGroup Command Parameters	32
Table 3.27	ModifyLunGroup Command Parameters	33
Table 3.28	AddWWNForLunGroup Command Parameters	33
Table 3.29	DeleteWWNForLunGroup Command Parameters	34
Table 3.30	AddLun Command Parameters	35

Table 3.31	DeleteLUN Command Parameters	36
Table 3.32	AddWWNForHostStorageDomain Command Parameters.	36
Table 3.33	DeleteWWNForHostStorageDomain Command Parameters	37
Table 3.34	DeleteWWN Command Parameters	37
Table 3.35	AddWWNGroup Command Parameters.	38
Table 3.36	ModifyWWNGroup Command Parameters.	38
Table 3.37	DeleteWWNGroup Command Parameters.	39
Table 3.38	AddWWNFor LUN Command Parameters	39
Table 3.39	DeleteWWNFor LUN Command Parameters	40
Table 3.40	AddHost Command Parameters	41
Table 3.41	DeleteHost Command Parameters	41
Table 3.42	ModifyHost Command Parameters	42
Table 3.43	GetHost Command Parameters	42
Table 3.44	AddHostInfo Command Parameters	43
Table 3.45	DeleteHostInfo	43
Table 3.46	ModifyHostInfo Command Parameters	44
Table 3.47	GetHostInfo Command Parameters	45
Table 3.48	GetLogFile Command Parameters	46
Table 3.49	ModifyDebugLevel Command Parameters	46
Table 3.50	GetAlerts Command Parameters	47
Table 3.51	DeleteAlerts Command Parameters	47
Table 3.52	AddURLLink Command Parameters	48
Table 3.53	DeleteURLLink Command Parameters	48
Table 3.54	GetURLLink Command Parameters.	48
Table 6.1	General Troubleshooting Information	105

Chapter 1 HiCommand™ Device Manager Overview

1.1 Overview of HiCommand™ Device Manager

HiCommand™ Device Manager consolidates storage operations and manages capacity for several models of RAID storage arrays, including Hitachi Freedom Storage™ and some Sun™ StorEdge™ disk arrays.

The Command Line Interface (CLI) version of the Device Manager software is available for expert users who prefer a character-based interface to create their own automation scripts.

- Targeted for users managing multiple storage arrays in open or shared environments, Device Manager quickly discovers the key configuration attributes of storage systems.
- HiCommand™ Device Manager provides user-defined hierarchical group management for disk storage.
- Device Manager allows users to begin proactively managing complex and heterogeneous storage environments quickly and effectively using an easy-to-use browser-based graphical user interface (GUI).
- Device Manager allows users to perform important operations such as adding/deleting storage, assigning logical unit numbers (LUNs), configuring fibre-channel ports, creating custom-size volumes, and managing LUN security.
- Device Manager gives storage administrators easier access to the existing array configuration, monitoring, and data management features.
- Device Manager enables remote storage management over secure IP connections and does not have to be direct-attached to the storage system.
- Device Manager displays detailed information on the configuration of the storage arrays added to the HiCommand™ Device Manager Server and provides subsystem alert presentation.
- Device Manager's built-in report facility compiles and presents key information in preformatted reports (HTML) and as comma-separated values for export.

HiCommand™ Device Manager is a component of the HiCommand™ family of software products, Hitachi Data Systems' storage management framework.

- Device Manager provides a consistent, easy-to-use, and easy-to-configure set of interfaces for managing storage from a common GUI over an IP connection.
- The Device Manager system includes the HiCommand™ Device Manager Server(s), the storage array(s) connected to the Servers, and the HiCommand™ Device Manager clients.
- The HiCommand™ Device Manager Web Client provides a web-distributed client for real-time interaction with the storage arrays being managed. You can also monitor and display volume usage statistics (optional).

Designed as an open framework, Device Manager provides a set of application programming interfaces (APIs) that allow industry-leading software vendors to seamlessly integrate their applications with HiCommand™ Device Manager.

- Software vendors are writing to the Device Manager APIs, including Sun™ Microsystems, VERITAS™, BMC®, Computer Associates™, and InterSAN™.
- Users can also “plug in” existing or new applications to the Device Manager system.
- Device Manager provides ‘link-and-launch’ integration with the Sun™ StorEdge Resource Management Suite software (SRM).
- Please contact your Hitachi Data Systems account team for the latest information on HiCommand™ Device Manager development partners.

The use of the HiCommand™ Device Manager product and all Hitachi Data Systems products is governed by the terms of your license agreement(s) with Hitachi Data Systems.

The use of the Sun™ StorEdge™ T3 array and all Sun™ products is governed by the terms of your license agreement(s) with Sun Microsystems.

1.2 HiCommand™ Device Manager Software Components

The HiCommand™ Device Manager system consists of the following software components:

- **Server:** The HiCommand™ Device Manager Server is LAN-attached to the storage arrays and controls HiCommand™ Device Manager operations based on requests from the HiCommand™ Device Manager clients. The HiCommand™ Device Manager clients (for example, CLI, Web Client, third-party application) communicate with the HiCommand™ Device Manager Server. The HiCommand™ Device Manager Agent sends information to the HiCommand™ Device Manager Server.

Note: For further information on the HiCommand™ Device Manager Server, please refer to the *HiCommand™ Device Manager Server Installation and Configuration Guide* (MK-91HC002).

- **Clients:**

- **Command Line Interface (CLI).** The HiCommand™ Device Manager CLI enables you to perform HiCommand™ Device Manager operations by issuing commands from the system command line prompt. The HiCommand™ Device Manager CLI communicates with and runs as a client of the HiCommand™ Device Manager Server.

This document describes and provides instructions for installing and using the HiCommand™ Device Manager CLI software.

- **Web Client.** The HiCommand™ Device Manager Web Client provides a web-based user interface for HiCommand™ Device Manager functions. The Web Client is a stand-alone Java™-based application that is deployed using the Java™ Web Start software. The Web Client communicates with and runs as a client of the HiCommand™ Device Manager Server.

Note: For further information on the HiCommand™ Device Manager Web Client, please refer to the *HiCommand™ Device Manager Command Web Client User's Guide* (MK-91HC001).

- **Third-party application.** HiCommand™ Device Manager supports third-party applications by providing an application program interface (API) that developers can use to interface with the HiCommand™ Device Manager Server. For details, please contact Hitachi Data Systems.
- **Agent (optional).** The HiCommand™ Device Manager Agent™ runs on the host server that is attached to one or more storage subsystems, collects data on the configuration and utilization of the attached storage, and sends this information to the HiCommand™ Device Manager Server. The HiCommand™ Device Manager clients display this information for the user when it is available on the Server. For further information on the Agent, please refer to the *HiCommand™ Device Manager Agent™ Installation Guide* (MK-91HC019).

Chapter 2 Requirements and Specifications

2.1 Requirements for CLI Operations

The system requirements for HiCommand™ Device Manager CLI operations are:

- **Storage Subsystems.** Before adding a storage subsystem to HiCommand™ Device Manager using the CLI (or other client), make sure that the subsystem is configured for HiCommand™ Device Manager operations. See section 2.2 for subsystem configuration requirements.
- **HiCommand™ Device Manager Server.** The HiCommand™ Device Manager Server must be installed, configured, and fully operational. Please refer to the *HiCommand™ Device Manager Server Installation and Configuration Guide* (MK-91HC002).
- **Platform for HiCommand™ Device Manager CLI.** HiCommand™ Device Manager 2.3 supports the following operating systems for the CLI:
 - Microsoft® Windows® 2000 Professional, Server, or Advanced Server with Service Pack 2 or later
 - Microsoft® Windows NT® Workstation or Server 4.0 with Service Pack 6a or later
 - Sun™ Solaris™ 8, Sun™ Solaris™ 9, or SPARC® platform only

Note: See section 2.3 for important information on using CLI release 2.0 with HiCommand™ Device Manager 2.1 and higher.

- **Agent (optional).** The HiCommand™ Device Manager Agent is not required for HiCommand™ Device Manager operations. If Agent is not installed, HiCommand™ Device Manager does not display the storage usage and file system statistics. For further information on the Agent, please refer to the *HiCommand™ Device Manager Agent Installation Guide* (MK-91HC019).

2.2 Storage Subsystem Requirements

Important: Before a storage subsystem can be added to the HiCommand™ Device Manager Server using the CLI (or other client), make sure that the subsystem is properly configured for HiCommand™ Device Manager operations.

The storage subsystem requirements are the same for all HiCommand™ Device Manager clients. Please refer to the *HiCommand™ Device Manager Web Client User's Guide* for detailed information on storage subsystem requirements. These important requirements are specified in only one user document, and are not repeated herein.

2.3 Restrictions of CLI Compatibility

2.3.1 R2.0 CLI with HiCommand™ Device Manager 2.0

The CLI module includes the R2.0 CLI module, which is located in the following directory:

<HiCommand Device Manager Install Directory>/HiCommandCLI/legacy/R2.0

Please use this R2.0 CLI module for HiCommand™ Device Manager 2.0 operations.

2.3.2 R2.0 CLI with HiCommand™ Device Manager 2.1 (and higher)

The following changes have been made to the CLI commands in HiCommand™ Device Manager Server Release 2.1 or higher and the CLI Release 2.0 combination.

1. **AddObjectForLogicalGroup**
Change the parameter "groupelement" object ID to "hoststoragedomain" (replacing "ports" and "paths"). See Table 2.1.
2. **DeleteObjectForLogicalGroup**
Change the parameter "groupelement" object ID to "hoststoragedomain" (replacing "ports" and "paths"). See Table 2.2.

Table 2.1 AddObjectForLogicalGroup Examples

Server Release	CLI Example
R2.0	HiCommandCLI AddObjectForLogicalGroup "objectid=ObjectID of LogicalGroup" "groupelements=Objectid of Port, Objectid of Path"
R2.1	Case: No HostStorageDomain of the target Port(1). (1) Creating the Port's HostStorageDomain by R2.1 CLI. For information on creating the Port's HostStorageDomain, see section 3.2.4. (2) Creating Path(on (1)'s HostStorageDomain) by R2.1 CLI. For information on creating a Path, see section 3.4.3. (3) Execute the following: "HiCommandCLI AddObjectForLogicalGroup "objectID= ObjectID of LogicalGroup " "groupelements= ObjectID of (1)'s HostStorageDomain"" Case: There is a HostStorageDomain of the target Port. (1) Getting ObjectID of the Port's HostStorageDomain. For information on getting the ObjectID of the Port's HostStorageDomain, see section 3.2.2. (2) Creating Path (on (1)'s HostStorageDomain) by R2.1 CLI. For information on creating a Path, see section 3.4.3. (3) Execute the following: "HiCommandCLI AddObjectForLogicalGroup "objectID= ObjectID of LogicalGroup " "groupelements= ObjectID of (1)'s HostStorageDomain"" Warning: When (1)'s HostStorageDomain have other paths, they are related to the LogicalGroup.

Table 2.2 DeleteObjectForLogicalGroup Examples

Server Release	CLI Example
R2.0	HiCommandCLI DeleteObjectForLogicalGroup "objectID= ObjectID of LogicalGroup " "groupelements= ObjectID of Port, ObjectID of Path "
R2.1	(1) Getting ObjectID of HostStorageDomain which is obtained the target LogicalGroup by R2.1 CLI. For information on getting ObjectID of HostStorageDomain which is obtained the target LogicalGroup, see section 3.3.4. (2) Getting ObjectID of HostStorageDomain that has the target Path by R2.1 CLI. For information on getting ObjectID of HostStorageDomain which has the target Path, see section 3.2.2. (3) Finding ObjectID of HostStorageDomain that has the specified Prot and Path. Searching HostStorageDomain ObjectID in (3)'s result which is in (1)'s result. (4) Execute the following: "HiCommandCLI AddObjectForLogicalGroup "objectID= ObjectID of LogicalGroup " "groupelements= ObjectID of (3)'s HostStorageDomain "" Warning: After (4), if (3)'s HostStorageDomain has other paths, they are not related to the LogicalGroup.

2.4 CLI Command Syntax

The CLI application provides a simple means of submitting requests to the HiCommand™ Device Manager Server and examining the Server's response. The application is run from the command line. Each execution of the application sends one request to the Server and receives the response to that request.

2.4.1 Launching CLI

For Windows® systems, launch the Command Line Interface by entering the following command:

```
C:\HiCommand\HiCommandCLI
```

For UNIX® systems, launch the Command Line Interface by entering the following command:

```
bash-2.04$ ./HiCommandCLI.sh
```

Note: If you launch the CLI without specifying any command line options, command arguments are displayed.

2.4.2 Format for the Command Line Values

The general format for the command line values is (see Table 2.3):

```
C:\HiCommand> HiCommandCLI [URL] [command] [options] [parameters]
```

- **URL:** see section 2.5.
- **command** and **parameters:** see section 3.2.
- **options:** see Table 2.4.

Table 2.3 HiCommand™ Device Manager CLI Variable Syntax

Variable	Function
URL	The complete URL of the HiCommand™ Device Manager Server, such as: http://localhost:2001/service . The URL contains the service address, unless the HiCommand™ Device Manager Server has been configured with a non-standard servlet alias. As described below, if you use a property file, it is possible to omit the URL.
command	The requested HiCommand™ Device Manager Server command, such as: GetStorageArray or AddLogicalGroup . The command is not required when reading the request from an XML file.
options	Options control the behavior of the CLI application. The options are entered in the typical UNIX fashion: each option has a single character representation and an alternate word representation; for the single character, prefix with a single dash; for the word, prefix with two dashes. For example: -t or -- messageTrace . Most options require an argument that follows.
parameters	Zero or more parameters, which are passed to the Server as part of the request. The required and optional parameters depend on which command is requested. Each parameter is represented as a name/value pair, for example, name=value .

Table 2.4 CLI Application Options

Option	Argument	Description
-u or --user	username	A valid login name for HiCommand™ Device Manager Server
-p or --password (see note)	<i>password</i> or <i>@password</i>	Login password for HiCommand™ Device Manager Server, or file which contains the login password (when preceded by a '@'). See note below.
-s or --secure	no argument	Use a secure connection (https) to communicate with the HiCommand™ Device Manager Server.
-i or --input	filename	Take input from the specified file, which must contain the HiCommand™ Device Manager Server request as a complete XML document. When this option is selected, the output of the Server's response is a XML formatted document.
-o or --output	filename	Send output to the specified file, instead of the console.
-t or --messagetrace	no argument	Output the request and the response, in the XML format sent and received, in the file MessageTrace.log.

Note: If you are specifying a HiCommand™ Device Manager Server **password** either on the command line or in the properties file, the password can be contained in a text file. Access to this file is generally restricted, but the application user must have access to this file. The file path is specified either relative to the location where the application is run or as an absolute path. The "@" character must precede the file path to indicate that it is a file and not the password itself. The text in the first line of the specified file is read and used as the password.

2.5 HiCommand™ Device Manager Server URL

The URL of the HiCommand™ Device Manager Server must be specified, either on the command line or in the properties files (described below). The URL must include the following items:

- Http protocol: `http://` or `https://` (see section 2.5.1 for https requirements)
- Host name or IP address of the HiCommand™ Device Manager Server
- Port number of the HiCommand™ Device Manager Server (default configuration is **2001**)
- Base address of the HiCommand™ Device Manager Server (default is **service**, unless the Server has been configured with a non-standard servlet alias).

2.5.1 Using CLI with HTTPS

To implement CLI while running HTTPS on the HiCommand™ Device Manager Server:

1. Download the HiCommandCerts file from the HiCommand™ Server using your browser:

`http://hcurl:2001/service/HiCommandCerts`

Note: `hcurl:2001` = HiCommand™ Server's host name and port.

2. Store the downloaded file in the local directory of the HiCommand™ CLI (the directory containing the HiCommandCLI.bat file).

Note: Use the filename `HiCommandCerts`.

3. Windows® systems only:

- a) Edit a copy of the `HiCommandCLI.bat` file using a new filename (to preserve the original file).
- b) Modify the `java` command as shown in Figure 2.1.

Note 1: Modify the `HiCommandCerts` file path to match the location of your HiCommandCLI directory.

Note 2: Use the `https` protocol for the HiCommand™ Device Manager Server URL.

Note 3: Use the `https` port for the HiCommand™ Device Manager Server URL (2443, unless configured differently in the Server's configuration file).

Note 4: Use the `-s` (or `--secure`) switch as shown in Figure 2.2.

Figure 2.2 illustrates a newly created batch file command line. The batch filename is `securecli.bat`.

```
if not defined HDVM_CLI_MEM_SIZE (
set HDVM_CLI_MEM_SIZE=256m )
java -Xms2m -Xmx%HDVM_CLI_MEM_SIZE% -cp HiCommandCLI.jar;jsse.jar \      -
Djavax.net.ssl.trustStore="c:/HiCommandCLI/HiCommandCerts" \
-Djava.protocol.handler.pkgs="com.sun.net.ssl.internal.www.protocol" \
com.hitachi.hds.cli.Hicmd %*
```

Figure 2.1 Modified java Command

Note: The backslashes (\) in Figure 2.1 are for legibility. In the batch file, the command is all on the same line.

```
securecli -s https://hcurl:2443/service GetServerInfo
```

Figure 2.2 Newly Created Batch File Command Line

2.6 CLI Return Responses

The HiCommand™ Device Manager CLI returns a variable at the end of its process. Table 2.5 lists and describes the CLI return variables.

Table 2.5 Return Variables

Return Variable	Description
0	CLI ends normally.
1	CLI found errors within its process (for example, parameter error).
2	CLI found errors in the HiCommand™ Device Manager Server.

2.7 Displaying CLI Help

If you want to display basic help information, launch CLI with no associated parameters. This basic information will include the current CLI version.

- For Windows® systems, the following command displays basic help information including the CLI version: `C:\>HiCommandCLI`
- For UNIX® systems, the following command displays basic help information including the CLI version: `bash-2.04$ ./HiCommandCLI.sh`

Figure 2.3 shows an example of basic help information, including the CLI version.

```
HiCommandCLI version "2.3"
USAGE: HiCommandCLI {server-url} {server-command} [options] [parameters]
AVAILABLE COMMANDS:
    AddStorageArray
    GetStorageArray
    DeleteStorageArray
    AddLunScan
    AddLogicalGroup
    DeleteLogicalGroup
    ModifyLogicalGroup
    GetLogicalGroup
    AddObjectForLogicalGroup
    DeleteObjectForLogicalGroup
    AddLunGroup
    ModifyLunGroup
    AddWwnForLUNGroup
    DeleteWwnForLUNGroup
    AddWwnGroup
    ModifyWwnGroup
    DeleteLunGroup
    AddLun
    DeleteLun
    AddHostStorageDomain
    DeleteHostStorageDomain
    AddWwnForLun
    DeleteWwnForLun
    AddWwnForHostStorageDomain
    DeleteWwnForHostStorageDomain
    DeleteWwn
    DeleteWwnGroup
    AddHost
    DeleteHost
    ModifyHost
    GetHost
    AddHostInfo
    DeleteHostInfo
    ModifyHostInfo
    GetHostInfo
    GetServerInfo
    GetLogFile
    ModifyDebugLevel
    GetDebugLevel
```

Figure 2.3 Example of Basic Help Information for CLI (continues on the next page)

```

GetAlerts
DeleteAlerts
AddArrayGroup
DeleteArrayGroup
AddLogicalUnit
DeleteLogicalUnit
ModifyLogicalUnit
AddSpareDrive
DeleteSpareDrive
ModifyPort
ModifyPortController
AddURLLink
DeleteURLLink
GetURLLink
RefreshStorageArrays
AddLUSE
DeleteLUSE

FOR COMMAND-SPECIFIC HELP, TYPE: "HiCommandCLI help {server-command}"

AVAILABLE OPTIONS:
  -u {username} or --user {username) login name for HiCommand Server

  -p {password} or --password {password} login password for HiCommand Server

  -s or --secure                               presence indicates secure
                                              connection (https)

  -i {filename} or --input {filename}          take input from the specified file,
                                              which must contain the complete xml
                                              request

  -o {filename} or --output {filename}          send output to the specified file,
                                              instead of the console

  -t or --messagetrace                          record raw XML request and response
                                              in MessageTrace.log file

SPECIFYING PARAMETERS:
  Specify parameters for a command using name/value pairs,
  like: serialnum=30051. Use the command-specific help to see
  the parameters for a given command.

-- BROWSE THE readme.html FILE FOR MORE DETAILED INSTRUCTIONS --

```

Figure 2.3 Example of Basic Help Information for CLI (continued from the previous page)

2.8 CLI Memory Requirements

To ensure successful CLI execution, the environment variable **HDVM_CLI_MEM_SIZE** must be set to an appropriate value. The default value is **256MB**. This value is specified in bytes and must be a multiple of 1024, and greater than 2MB (**k** or **K** to indicate kilobytes, **m** or **M** to indicate megabytes). For example, "5000k" means 5000 kilobytes, and "100000" means 100000 bytes.

Get commands such as **GetstorageArray**, **GetLogicalGroup**, and **GetHostInfo** may require more memory due to command parameters assigned. Figure 2.4 shows an example of executing the command **GetstorageArray** in Windows® systems:

```
HiCommandCLI GetstorageArray subtarget=LogicalUnit ^  
"lusubinfo=Path,LDEV" "pathsubinfo=WWN,WWNGroup,HostInfo"
```

Figure 2.4 GetstorageArray in Windows® Systems

Note: If the HiCommand™ Device Manager Server is managing multiple subsystems, CLI memory requirements will be increased. Make sure to contact your Hitachi Data Systems representative for further information.

Note: If the value of **HDVM_CLI_MEM_SIZE** is less than what is required by a CLI command, CLI will terminate, and the following error message will be displayed:

```
Exception in thread "main" java.lang.OutOfMemoryError  
<<no stack trace available>>
```


Chapter 3 Specifying Command Parameters

3.1 Description of Syntax

The parameters that form the input to a command are specified on the command line as name/value pairs. The parameter name is specified first, then the equal sign and the parameter value, as follows:

```
ipaddress=192.168.2.2
```

Whenever a parameter value requires one or more space characters, double quotes can be used to enclose the parameter value, as follows:

```
description="Our Newest Array"
```

Each command may have required parameters, optional parameters, or both. Any parameter can be specified in the application's properties file. For more information on HiCommand™ Device Manager properties, please refer to the *HiCommand™ Server Installation and Configuration Guide* (MK-91HC002). For more information on HiCommand™ Device Manager client operations (for example, Web Client, CLI, third-party applications), please refer to the *HiCommand™ Device Manager Web Client User's Guide*. If a parameter is specified both on the command line and in the properties file, the value from the command line is used. The value of the parameter specified in the properties file is used when the parameter is either a required or an optional parameter for the requested command, *and* the parameter is not specified on the command line.

The CLI includes a help file. To see the parameters associated with a particular command, enter:

```
C:\HiCommand> HiCommandCLI help command-name
```

For example, to display the output shown in Figure 3.1, enter:

```
C:\HiCommand> HiCommandCLI help AddLun
```

```
COMMAND: AddLun
DESCRIPTION: Defines a path from a host to a volume.
PARAMETERS:
    serialnum    (required) Serial Number of the Storage Array of the new Path
    model        (required) Model of the Storage Array of the new Path
    name         (optional) Name of the new Path
    port         (required) Port Number of the new Path
    domain       (required) DomainID of the new PathFor 9900V and 9500V (with
                        LUNManagement), required; others, optional. Domain Number
                        of the new Path, cannot be set with parameter wwn
                        together.
    scsi         (required) SCSI ID of the new Path
    lun          (required) Number of the actual LUN used for path
    devnum       (optional) Device number used to identify this logical unit
    lusedevnums  (optional) "Comma-separated list of LDEV devnums required for
                        LUSE LUN definition
    wwn          (optional) Except 9900V and 9500V (with
                        LUNManagement).Comma-separated list of WWN to secure the
                        path, can not be set with parameter domain together.
```

Figure 3.1 Example of CLI Help for AddLun Command

3.2 Storage Array Commands

For information on storage array operations, please refer to the *HiCommand™ Device Manager Web Client User's Guide* (MK-91HC001).

3.2.1 AddStorageArray

AddStorageArray performs a search for storage arrays, populating the HiCommand™ Device Manager database with the identified device elements (see Table 3.1).

Table 3.1 AddStorageArray Command Parameters

Parameter Name	Required	Description
ipaddress	Required	IP address of the array
family	Required	Array family of the target array
displayfamily	Optional	Display array family of the target array
searchcommunity	Optional (9900 only)	Search community, used for identifying device elements via SNMP (9900 array). Default is public.
ipaddress2	Optional (9200, 9500V only)	Second IP Address for array (9200 and 9500V arrays)
userid	Optional (9200, 9500V, 9900V, T3 only)	User name for array access (9900V, 9200, 9500V and T3 arrays). For 9200 and 9500V, required only at initial identification of those arrays with Password Protection feature enabled, optional for other cases. For 9900V and T3, required only at the initial identification.
arraypasswd	Optional (9200, 9500V, 9900V, T3 only)	User password for array access (9900V, 9200, 9500V and T3 arrays). For 9200 and 9500V, required only at initial identification of those arrays with Password Protection feature enabled, optional for other cases. For 9900V and T3, required only at the initial identification.

3.2.2 GetStorageArray

GetStorageArray returns information on the storage array (see Table 3.2).

Note: Subtarget parameters are only valid when specified with the indicated value. For subtargets **Component**, and **CommParameters**, no additional parameters are applicable.

Table 3.2 GetStorageArray Command Parameters (continues on the next page)

Parameter Name		Required	Description
serialnum		Optional (omit to include all storage arrays)	Serial number of the storage array
model		Optional (omit to include all storage arrays)	Model of the storage array
subtarget		Optional (omit to return only storage arrays with none of the specified devices)	Specifies the type of elements to return within each array. Possible values include Component, Path, Port, LogicalUnit, LDEV, PDEV, ArrayGroup, PortController, FreeSpace, Filter or CommParameters, HostStorageDomain, and ReplicationInfo, which are not case-sensitive.
subtarget =Path	objectid	Optional (omit to include all paths of the array(s))	Object ID of the Path of the array to return; a specific array must be specified by serialnum and model.
	pathsubinfo	Optional	More information on the Path(s) to return. Possible values are: LogicalGroup,WWN, WWNGroup, and HostInfo, which are not case-sensitive.
subtarget =Port	port	Optional (omit to include all ports of the array(s))	ID of the port of the array to return; a specific array must be specified by serial number and model.
	portsubinfo	Optional	More information on the port(s) to return. Possible values are: WWN, WWNGroup, LUNGroup, and HostStorageDomain, which are not case-sensitive..
subtarget =Logical Unit	objectid	Optional (omit to include all LogicalUnits of the array(s))	Object ID of the LogicalUnit of the array to return; a specific array must be specified by serialnum and model.
	lufilter	Optional	Filter for selecting LogicalUnit to return. Possible values are: ALL, ASSIGNED, FREE, which are not case-sensitive. If this parameter is not specified or this value is "ALL", all LUs are returned. "FREE" returns those LogicalUnits that do not have LUNs (Paths) assigned to them. "ASSIGNED" returns just those LogicalUnits that have LUNs assigned to them.
	lusubinfo	Optional	More information on the LogicalUnit(s) to return. Possible values are: Path, LDEV, which are not case-sensitive..
	pathsubinfo	Optional	More information about the Path on the LogicalUnit(s) to return. Path must be specified by lusubinfo. Possible values are: WWN, WWNGroup, and HostInfo, which are not case-sensitive.
subtarget=LDEV (parameter is devnum)		Optional (omit to include all LDEVs of the array(s))	Device number of the LDEV of the array to return; a specific array must be specified by serialnum and model.

Table 3.2 GetStorageArray Command Parameters (continued)

Parameter Name		Required	Description
subtarget=PDEV	pdevid	Optional (omit to include all PDEVs of the array(s))	Right-most digit of the object PDEV ID of the PDEV of the array to return; PDEV ID is the right-most digit in the object ID (Object ID consists of 4 parts separated by a dot.) A specific array must be specified by serialnum and model.
	arraygroupobjid	Optional	Object IDs of the ArrayGroup PDEVs within this ArrayGroup are returned. Set ALL to get PDEVs within all ArrayGroups. If it is not ALL, a specific array must be specified by serialnum and model.
subtarget=ArrayGroup	objectID	Optional (omit to include all array groups of the array(s))	Object ID of the array group of the array to be returned; a specific array must be specified by serialnum and model parameters.
subtarget=PortController	controllernum	Optional (omit to include all port controllers of the array(s))	Controller number of the port controller of the array to return; a specific array must be specified by serialnum and model parameters.
	controllersubinfo	Optional	More information on the PortController (s) to be returned. Possible values are: IPAddress and/or PairedPortController, which are not case-sensitive.
subtarget=FreeSpace	arraygroupobjid	Optional (omit to include all FreeSpace of the array(s))	Object ID of the array group that includes free space to be returned; a specific array must be specified by serialnum and model parameters.
subtarget=Filter	objectid	Required	Object ID of the child element of the array. The array that contains the object ID specified by this parameter is returned.
subtarget=HostStorageDomain	port	Optional	Port ID of the Host Storage Domain.
	domain	Optional	Domain ID of the Host Storage Domain.
	hsdsbinfo	Optional	More information on the HostStorageDomain(s) to return. Possible values are: WWN, Path, and FreeLUN, which are not case-sensitive.
subtarget=ReplicationInfo	objectid	Optional (omit to include all replication information of the array(s))	ReplicationInfo array Object ID. The array is specified by the parameters serialnum and model.
	replicationsubinfo	Optional	More information on the ReplicationInfo to be returned. Only possible value is LogicalUnit, which is not case-sensitive.

3.2.3 DeleteStorageArray

DeleteStorageArray removes a Hitachi storage array from the HiCommand™ Device Manager database, but does not attempt to change the configuration of any storage device (see Table 3.3).

Table 3.3 DeleteStorageArray Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array to delete
model	Required	Model of the storage array to delete

3.2.4 AddHostStorageDomain

AddHostStorageDomain adds a HostStorageDomain. For the 9900V Series, it creates a HostGroup in the subsystem (see Table 3.4).

Table 3.4 AddHostStorageDomain Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array for the Host Storage Domain.
model	Required	Model of the storage array for the Host Storage Domain.
port	Required	Port ID of the Host Storage Domain (a specific array must be specified by serial number and model).
domain	Required	DomainID of the Host Storage Domain.

3.2.5 DeleteHostStorageDomain

DeleteHostStorageDomain deletes a HostStorageDomain. For the 9900V Series, it deletes a HostStorageDomain (HostGroup) from the subsystem (see Table 3.5).

Table 3.5 DeleteHostStorageDomain Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array for the Host Storage Domain.
model	Required	Model of the storage array for the Host Storage Domain.
port	Required	The port ID of the Host Storage Domain to delete.
domain	Required	The domain ID of the Host Storage Domain to delete.
deletionoption	Optional (omit to delete luse)	Only possible value is: "lusekeep"

3.2.6 AddArrayGroup

AddArrayGroup adds an array group to a storage device (see Table 3.6). An array group consists of a number of physical disk drives that are managed as logical devices according to the RAID level. The storage array is returned with the new array group enclosed.

Note: This function is not available for the Lightning™ 9900V or 9900 subsystem.

Table 3.6 AddArrayGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array for the array group
model	Required	Model of the storage array for the array group
chassis	Required	Number of chassis for the array group
raidtype	Required	Raid type of the array group
pdevnums	Required	Comma-separated list of physical disk (PDEV) IDs (see pdevid in Table 3.2 for more information)

3.2.7 DeleteArrayGroup

DeleteArrayGroup deletes an array group to a storage device (see Table 3.7). When an array group is deleted, all LogicalUnits on that array group are deleted.

Note: This function is not available for the Lightning™ 9900V or 9900 subsystem.

Table 3.7 DeleteArrayGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array for the array group
model	Required	Model of the storage array for the array group
chassis	Required	Number of chassis for the array group
groupnum	Required	Group number of the group array to delete

3.2.8 AddSpareDrive

AddSpareDrive adds a spare drive to a storage device (see Table 3.8). The spare drive is a physical disk drive that is used by the storage device as a backup or standby. The spare drive is returned within a storage array.

Note: This function is not available for the Lightning™ 9900V or 9900 subsystem.

Table 3.8 AddSpareDrive Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array where the spare drive is added
model	Required	Model of the storage array where the spare drive is added
pdevid	Required	Physical device ID for the new spare drive (see pdevid in Table 3.2 for more information)

3.2.9 DeleteSpareDrive

DeleteSpareDrive deletes or releases a spare drive from a storage device (see Table 3.9). The physical device currently used as a spare is added to the pool of unused devices.

Note: This function is not available for the Lightning 9900V or 9900 subsystem.

Table 3.9 DeleteSpareDrive Command Parameters

Parameter Name	Required	Description
serialnum	Required	serial number of the storage array of the spare drive
model	Required	model of the storage array of the spare drive
pdevid	Required	Physical device ID of the spare drive (See pdevid in Table 3.2 for more information)

3.2.10 AddLogicalUnit

AddLogicalUnit creates an LDEV and LogicalUnit in HiCommand™ (see Table 3.10). For the 9900V and 9900, an LDEV (mainframe or open) is created, as well as an LU by the SVP (open volumes only). For the 9200, 9500V and T3, an LU is created (and HiCommand™ Device Manager creates a corresponding pseudo-LDEV). The new LogicalUnit is returned within a storage array.

Table 3.10 AddLogicalUnit Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array where LU is added.
model	Required	Model of the storage array where the LU is added.
chassis	Required	Number of chassis for the array group where LU is added.
groupnum	Required	Group number of the array group where LU is added.
capacity	Required (9900V, 9900, 9500V and 9200)	Desired capacity (in KB) for the new LU.
defportcntl	Required (9500V and 9200)	Number of the port controller for the LU, if not present, will use array group emulation.
emulation	Optional (9900V and 9900)	Open emulation for the LU.
lusubinfo	Optional (Do not include any LU LDEVs)	If set to LDEV, return the information on LDEV(s) related to the LU. The only value possible is LDEV, which is not case-sensitive.

3.2.11 DeleteLogicalUnit

DeleteLogicalUnit deletes the LogicalUnit and corresponding LDEV from the storage device (see Table 3.11).

Table 3.11 DeleteLogicalUnit Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array of the LU.
model	Required	Model of the storage array of the LU.
devnum	Required	Device number of the LU.

3.2.12 ModifyLogicalUnit

ModifyLogicalUnit modifies the LU and its corresponding LDEV in a storage device (see Table 3.12).

Note: the LU should have a Path.

Table 3.12 ModifyLogicalUnit Command Parameters

Parameter Name	Required	Description
Serialnum	Required	Serial number of the storage array of the LU.
Model	Required	Model of the storage array of the LU.
Devnum	Required	Device number of the LU.
commanddevice	Required	New setting for command device. "true" = commanddevice, "false" ≠ commanddevice.

3.2.13 AddLUSE

AddLUSE creates a LUSE without any path in the storage device (Table 3.13).

Table 3.13 AddLUSE Command Parameters

Parameter Name	Required	Description
Serialnum	Required	Serial number of the storage array from which LUSE is added.
Model	Required	Model of the storage array from which the LUSE is added.
Devnum	Required	Comma-separated list of LDEV device numbers to create the LUSE.

3.2.14 DeleteLUSE

DeleteLUSE deletes a LUSE without any path in the storage device (see Table 3.14).

Table 3.14 DeleteLUSE Command Parameters

Parameter Name	Required	Description
Serialnum	Required	Serial number of the storage array from which the LUSE is to be deleted.
Model	Required	Model of the storage array from which the LUSE is to be deleted.
Devnum	Required	Device number used to identify the LUSE to be deleted. LUSE device number must be valid in the storage array.

3.2.15 ModifyPort

ModifyPort supports changing characteristics of a Port (see Table 3.15).

Table 3.15 ModifyPort Command Parameters (continues on the next page)

Parameter Name	Required	Description
serialnum	Required	Serial number of the port's storage array.
model	Required	Model of the port's storage array.
port	Required	Number of the port.
topology	Optional	New topology value for the port (such as "FABRIC_ON_FCAL", "NOT_FIBRE", etc.).
fibreaddr	Optional	New fibre channel address for the port (the hex string representation).
channelspeed	Optional (9900V only)	New transmission speed of a fiber channel for the port (0:automatic, 1:1 GB/s, 2:2 GB/s).
lunsec	Optional	New setting for LUN security enabled (either "true" or "false").
portoption	Optional	Semicolon-separated new option list of the port. Possible values for 9200 are: Reset ALL LIP Port Mode Reset/LIP Mode (Process) Reset/LIP Mode (Signal) Reset Target (Reset Bus Device) Mode Reserve Mode Reset Logical Unit Mode Reset Logout of Third Party Process Mode Possible values for 9500V are: Reset ALL LIP Port Mode Reset/LIP Mode (Process) Reset/LIP Mode (Signal).
domain	Optional	DomainID of the Host Storage Domain (please refer to notes).

Table 3.15 ModifyPort Command Parameters (continued)

Parameter Name	Required	Description
hostmode	Optional	New host mode value of the port (such as "HP", "Netware", etc.).
hostmode2	Optional	New host mode value list of the port. Possible values for 9900 are: "Normal" "Sequent" "HP" "Solaris" "Netware" "PC Server" "Tru64" "HI-UX" "AIX" Possible values for 9900V are: "Normal" "Sequent" "HP" "Solaris" "Netware" "PC Server" "IO Trace" "HI-UX" "AIX" Possible values for 9200 and 9500V are: "Standard" "Open VMS" "Wolfpack" "TRESPASS" Possible values for T3 are: "Standard" "Sun"

Note: When the **hostmode** or **hostmode2** parameter is changed, a domain parameter must be specified.

3.2.16 ModifyPortController

ModifyPortController supports changing characteristics of a PortController (see Table 3.16).

Note: This function is not available for the Thunder™ 9200 and 9500V subsystem or Sun™ StorEdge™ T3 array.

Table 3.16 ModifyPortController Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the port controller's storage array.
model	Required	Model of the port controller's storage array.
ctrlid	Required	Number of the port controller.
adaptermode	Required	New adapter mode value of the port controller.

3.2.17 RefreshStorageArrays

RefreshStorageArrays re-discovers all discovered storage subsystems one time (see Table 3.17).

Table 3.17 RefreshStorageArrays Command Parameters

Parameter Name	Required	Description
interval	Optional	Interval time in minutes from storage subsystem re-discovery completion to start of next storage subsystem re-discovery. If it is not set, the default interval time is 0.

This command is designed to refresh configuration data for all discovered storage arrays managed by HiCommand™ Device Manager. Any storage array reconfigurations made by tools other than HiCommand™ Device Manager (e.g., 9900V Remote Console - Storage Navigator) will be reflected in the HiCommand™ Device Manager Configuration database.

If storage arrays are managed by the HiCommand™ Device Manager and other tools, you should schedule RefreshStorageArray periodically so that the HiCommand™ Device Manager Configuration database is kept current.

3.3 Logical Group Commands

When a group is created, the HiCommand™ Device Manager Server generates an object ID. In order to use a command that requires an object ID, the ID must be known. When a group is created, the Server returns the new group, including the ID. When all groups are retrieved with the **GetLogicalGroup** command, the ID of all groups is provided.

For further information on logical group operations, please refer to the *HiCommand™ Device Manager Web Client User's Guide* (MK-91HC001).

3.3.1 AddLogicalGroup

AddLogicalGroup creates a new logical group, which supports the organization and naming of related devices on the HiCommand™ Device Manager Server (see Table 3.18). The new group is returned, including the object ID.

Table 3.18 AddLogicalGroup Command Parameters

Parameter Name	Required	Description
groupname	Required	Name for the new logical group. This must be unique among groups within its parent (if the parent parameter is supplied) or unique among top-level groups (if no parent is specified).
iconfile	Required	Name of the icon file that visually represents the new logical group.
parent	Optional	Object ID of another logical group that contains this new group. This must be the valid ID of a group, and the parent must either contain other groups only or be empty.

3.3.2 DeleteLogicalGroup

DeleteLogicalGroup deletes an existing logical group from the HiCommand™ Device Manager Server (see Table 3.19).

Table 3.19 DeleteLogicalGroup Command Parameters

Parameter Name	Required	Description
objectid	Required	Object ID of the logical group to delete.

3.3.3 ModifyLogicalGroup

ModifyLogicalGroup changes one or more characteristics of an existing logical group (see Table 3.20).

Table 3.20 ModifyLogicalGroup Command Parameters

Parameter Name	Required	Description
objectid	Required	Object ID of the logical group to modify.
groupname	Optional (group name not modified if omitted)	New name for the specified logical group. This must be either unique among groups within its parent, or if not contained in another group, unique among top-level groups.
iconfile	Optional (group icon file not modified if omitted)	Name of the icon file that visually represents the new logical group.
parent	Optional (group's parent not modified if omitted)	Object ID of another logical group that contains the specified group. This must be the valid ID of a group and the parent must either contain only other groups or be empty.

3.3.4 GetLogicalGroup

GetLogicalGroup returns either a specified logical group or a list of all groups (see Table 3.21). By default, the characteristics of the group are provided, but not the group's contents. If you specify a subtarget, the group's contents of the specified types are returned. If a logical group contains other groups, the contained groups cannot be returned with the group. Instead, the relationship between containing and contained groups is determined by the parent attribute of contained groups.

Table 3.21 GetLogicalGroup Command Parameters

Parameter Name	Required	Description
objectid	Optional (omit to include all logical groups)	Object ID of the desired logical group.
subtarget	Optional (omit to return only logical groups(s), with no contained elements)	Comma-separated list specifying the contained elements to return. Possible values are " HostStorageDomain ", Path and/or " Host ", which are not case-sensitive.

3.3.5 AddObjectForLogicalGroup

AddObjectForLogicalGroup adds one or more existing objects (**HostStorageDomain** or host) to a specified logical group on the HiCommand™ Device Manager Server (see Table 3.22).

Table 3.22 AddObjectForLogicalGroup Command Parameters

Parameter Name	Required	Description
objectid	Required	Object ID of the logical group.
groupelements	Required	Comma-separated list of one or more Object IDs of the HostStorageDomains and/or hosts to add to the logical group.

3.3.6 DeleteObjectForLogicalGroup

DeleteObjectForLogicalGroup removes the specified object or objects from the logical group (see Table 3.23).

Table 3.23 DeleteObjectForLogicalGroup Command Parameters

Parameter Name	Required	Description
objectid	Required	Object ID of the logical group.
groupelements	Required	Comma-separated list of one or more object IDs of the HostStorageDomains and/or hosts to remove from the logical group. These must be valid IDs of objects currently in the group.

3.3.7 AddLunScan

AddLunScan scans a storage array for LUNs not assigned to a logical group (see Table 3.24). It assigns those LUNs to subgroups of the "LUN Scan" group according to port and security. **AddLunScan** creates those subgroups, as necessary, returning the specified storage array, and each of the previously unassigned LUNs.

Table 3.24 AddLunScan Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array to scan.
model	Required	Model of the storage array to scan.

3.4 LUN Commands

For further information on LUN operations, please refer to the *HiCommand™ Device Manager Web Client User's Guide* (MK-91HC001).

3.4.1 AddLunGroup

AddLunGroup adds a LUN Group to a Port of the target storage array (only 9900 series) and adds Paths to the LUN Group (see Table 3.35).

Table 3.25 AddLunGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the target storage array to which the LUN Group is added.
model	Required	Model of the target storage array to which the LUN Group is added.
port	Required	Port number of the LUN Group.
groupelements	Required	Comma-separated list of one or more Object IDs of the paths to be added to the LUN group.
nickname	Required	Unique nickname of the LUN Group.
name	Optional	The name given to the LUN Group.

3.4.2 DeleteLunGroup

DeleteLunGroup deletes a LUN Group assigned to a port; any LUNs (paths) in the group are not deleted (see Table 3.26). A WWN can be specified and access is removed for just that WWN.

Table 3.26 DeleteLunGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the LUN Group's storage array.
model	Required	Model of the LUN Group's storage array.
port	Required	ID for the port of LUN Group to delete.
lungroupid	Required	Object ID of the LUN Group to delete.

3.4.3 ModifyLunGroup

ModifyLunGroup modifies a LUN Group assigned to a port of the target storage array (only 9900 series) and adds Paths in the LUN Group (see Table 3.27).

Table 3.27 ModifyLunGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array in which the LUN Group exists
model	Required	Model of the storage array in which the LUN Group exists.
port	Required	Port number of the LUN Group.
objectid	Required	Object ID of the LUN Group.
groupelements	Required	Comma-separated list of one or more Object IDs of the paths to be added to the LUN group.
nickname	Optional	Unique nickname of the LUN Group
name	Optional	The name given to the LUN Group

3.4.4 AddWWNForLunGroup

AddWWNForLunGroup secures a LUNGroup by assigning one or more WWN(s) to it (see Table 3.28). If all of the WWNs in a WWNGroup are assigned, the WWNGroup is used to secure the LUNGroup. If any of WWNs in a WWNGroup are not assigned, this command will fail. Only new WWNs (none that pre-exist on the port) can be added.

Table 3.28 AddWWNForLunGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array in which the LUN Group exists
model	Required	Model of the storage array in which the LUN Group exists.
port	Required	Port number of the LUN Group.
objectid	Required	Object ID of the LUN Group.
wwn	Required	Comma-separated list of WWNs to secure the LUN group

3.4.5 DeleteWWNForLunGroup

DeleteWWNForLunGroup removes one or more WWN(s) from a LUNGroup (see Table 3.29).

Table 3.29 DeleteWWNForLunGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array in which the LUN Group exists
model	Required	Model of the storage array in which the LUN Group exists.
port	Required	Port number of the LUN Group.
objectid	Required	Object ID of the LUN Group.
wwn	Required	Comma-separated list of WWNs that exist in the LUN group.

3.4.6 AddLun

AddLun defines a path from a host to a volume (see Table 3.30). At the time the path is defined, a LUN Expansion (LUSE) logical unit can be specified.

Table 3.30 AddLun Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the new path's storage array.
model	Required	Model of the new path's storage array.
name	Optional	Name of the new path.
port	Required	Port number of the new path.
domain	Required: 9900V, 9500V (with LUNManagement), T3 Optional: all subsystems except 9900V, 9500V (with LUNManagement), T3	Domain ID of the new path. Cannot be set when using wwn parameter.
scsi	Required: all except T3 Optional: T3 (Ignored if present, assigned by T3)	SCSI ID of the new path.
lun	Required	Actual LUN used for new path.
devnum	Required: T3	Device number used to identify the new path (not provided when a LUSE is defined).
lusedevnums	Required: T3	Comma-separated list of LDEV device numbers to create a LUSE for the path (either devnum or lusedevnums must be provided, but not both).
wwn	Optional except for 9900V, 9500V (using LUN anagement), T3	Comma-separated list of WWN to secure the Path. Cannot be set when using domain parameter.

3.4.7 DeleteLun

DeleteLun deletes a path from a host to a volume (see Table 3.31). If the LUN is secured, the corresponding WWN is required.

Table 3.31 DeleteLUN Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the path's storage array.
model	Required	Model of the path's storage array.
port	Required	Port number of the path.
domain	Required	Domain ID of the path.
devnum	Required	Device number used to identify the path.
deletionoption	Optional(omit to delete luse)	Only value possible: "lusekeep", which is not case-sensitive.

3.4.8 AddWWNForHostStorageDomain

AddWWNForHostStorageDomain secures LUNs in a HostStorageDomain by assigning one or more WWNs to it (see Table 3.32).

Table 3.32 AddWWNForHostStorageDomain Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array for the Host Storage Domain.
model	Required	Model of the storage array for the Host Storage Domain.
port	Required	Port ID of the Host Storage Domain.
domain	Required	Domain ID of the Host Storage Domain.
wwn	Required	Comma-separated list of WWN to secure the Host Storage Domain.

3.4.9 DeleteWWNForHostStorageDomain

DeleteWWNForHostStorageDomain removes security for LUNs that are under a specified HostStorageDomain. The HostStorageDomain must already exist and be secured (see Table 3.33).

Table 3.33 DeleteWWNForHostStorageDomain Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array for the Host Storage Domain.
model	Required	Model of the storage array for the Host Storage Domain.
port	Required	Port ID of the Host Storage Domain.
domain	Required	Domain ID of the Host Storage Domain.
wwn	Required	Comma-separated list of WWN to unsecure the Host Storage Domain.

3.4.10 DeleteWWN

DeleteWWN removes a WWN from a port (see Table 3.34).

Table 3.34 DeleteWWN Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the port's storage array.
model	Required	Model of the port's storage array.
port	Required	Port number.
wwn	Required	WWN to delete from the port.

3.4.11 AddWWNGroup

AddWWNGroup adds a WWN Group to a target storage array Port (9900 subsystem only) and two or more WWN(s) to the WWN Group (Table 3.35). The maximum number of WWNs in a Port WWN Group is 127. Only new WWNs (none that pre-exist on the port) can be added.

Table 3.35 AddWWNGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the target storage array to which the WWN Group is added.
model	Required	Model of the target storage array to which the WWN Group is added.
port	Required	Port number of the WWN Group.
wwn	Required	Comma-separated list of WWNs to be added to the WWN group.
nickname	Required	A unique nickname of the WWN group.
name	Optional	The name given to the WWN group.

3.4.12 ModifyWWNGroup

ModifyWWNGroup modifies a WWN Group assigned to a target storage array Port (9900 subsystem only) and adds two or more WWN(s) to the WWN Group (Table 3.36).

Table 3.36 ModifyWWNGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the storage array in which the WWN Group exists.
model	Required	Model of the storage array in which the WWN Group exists.
port	Required	Port number of the WWN Group.
wwn	Required	Comma-separated list of WWNs to be added to the WWN group.
nickname	Optional	Unique nickname of the WWN Group.
name	Optional	The name given to the WWN group.

3.4.13 DeleteWWNGroup

DeleteWWNGroup removes a WWN Group from a port (see Table 3.37). The Corresponding WWNs are not removed from the port.

Table 3.37 DeleteWWNGroup Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the port's storage array.
model	Required	Model of the port's storage array.
port	Required	Port number.
wwngroup	Required	Name of the WWN group to delete from the port.

3.4.14 AddWWNForLun

AddWWNForLun secures a path by assigning a WWN to it. This command only supports the 9900, 9500V(LUNSecurity) and 9200 subsystems.

Table 3.38 AddWWNFor LUN Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the port's storage array.
model	Required	Model of the port's storage array.
port	Required	Port number.
domain	Required	Domain ID of the path.
devnum	Required	Device number used to identify the path.
wwn	Required	Comma-separated list of WWNs to secure the path.

3.4.15 DeleteWWNForLun

DeleteWWNForLun removes access to a WWN path (Table 3.39). This command only supports the 9200, 9900, 9500V Series (LUNSecurity) subsystems.

Table 3.39 DeleteWWNFor LUN Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the port's storage array.
model	Required	Model of the port's storage array.
port	Required	Port number.
domain	Required	Domain ID of the path.
devnum	Required	Device number used to identify the path.
wwn	Required	Comma-separated list of WWNs to secure the path.

3.5 Host Management Commands

The following commands support the management of hosts and their HostInfo records. Although typically host agents produce HostInfo records, they can be manually created by these commands. Some commands require a host's server-generated object ID.

For further information on host operations, please refer to the *HiCommand™ Device Manager Web Client User's Guide* (MK-91HC001).

3.5.1 AddHost

AddHost adds information about a host server to the HiCommand™ Device Manager Server (see Table 3.40).

Table 3.40 AddHost Command Parameters

Parameter Name	Required	Description
hostname	Required	Name of the new host.
wwnlist	Optional	Comma-separated list of WWNs.
ipaddress	Optional	IP address of the new host.

3.5.2 DeleteHost

DeleteHost Deletes information about a host server from the HiCommand™ Device Manager Server (see Table 3.41).

Table 3.41 DeleteHost Command Parameters

Parameter Name	Required	Description
objectid	Required	Object ID of the host to remove.

3.5.3 ModifyHost

ModifyHost modifies information about a host server (see Table 3.42).

Table 3.42 ModifyHost Command Parameters

Parameter Name	Required	Description
objectid	Required	Object ID of the host to modify.
hostname	Optional (omit if the name remains unchanged)	New name for the existing host.
wwnlist	Optional (If not present, All WWNs in the Host will be deleted)	Comma-separated list of WWNs to replace existing WWNs.
lppaddress	Optional (omit if the host IP address remains unchanged)	New IP address for the existing host.

3.5.4 GetHost

GetHost returns the information about one or all host servers (see Table 3.43).

Table 3.43 GetHost Command Parameters

Parameter Name	Required	Description
objectid	Optional (omit to include all hosts)	Object ID of the host to return.

3.5.5 AddHostInfo

AddHostInfo adds information host-based information on a LUN (see Table 3.44).

Table 3.44 AddHostInfo Command Parameters

Parameter Name	Required	Description
serialnum	Required	Serial number of the path's storage array.
model	Required	Model of the path's storage array.
hostname	Required	Name of the host.
ipaddress	Optional	IP address of the host.
mountpoint	Optional	Mount point on the host file system corresponding to the path.
port	Required	Port that the path is on.
domain	Required	Domain ID of the path.
devnum	Required	Device number of the path.
oscsibus	Required	Host SCSI bus number.
oscsiid	Required	Virtualized host SCSI ID.
oslun	Required	Virtualized host LUN.
portwwn	Optional	Port WWN on the host bus adapter.
filetype	Optional	Type of the File System.
filename	Optional	Name of the File System.
size	Optional	LUN size in MB.
percentused	Optional	Percent of the LUN in use.

3.5.6 DeleteHostInfo

DeleteHostInfo deletes a HostInfo instance from the HiCommand™ Device Manager Server (see Table 3.45).

Table 3.45 DeleteHostInfo

Parameter Name	Required	Description
objectid	Required	Object ID of the HostInfo record to remove.

3.5.7 ModifyHostInfo

ModifyHostInfo modifies a **HostInfo** record in the HiCommand™ Device Manager Server (see Table 3.46).

Table 3.46 ModifyHostInfo Command Parameters

Parameter Name	Required	Description
objectid	Required	Object ID of the HostInfo record to modify.
serialnum	Optional (omit when no change is required)	New serial number of the path's storage array.
model	Optional (omit when no change is required)	New model of the path's storage array.
hostname	Optional (omit when no change is required)	New name of the host.
ipaddress	Optional (omit when no change is required)	New IP address of the host.
mountpoint	Optional (omit when no change is required)	New mount point on the host file system corresponding to the path.
port	Optional (omit when no change is required)	New port on the path.
domain	Optional (omit when no change is required)	New domain ID of the path.
devnum	Optional (omit when no change is required)	New device number of the path.
portwwn	Optional (omit when no change is required)	New Port WWN on the host bus adapter.
filetype	Optional (omit when no change is required)	New type of the File System.
filename	Optional (omit when no change is required)	New name of the File System.
size	Optional (omit when no change is required)	New LUN size in MB.
percentused	Optional (omit when no change is required)	New percentage of the LUN in use.

3.5.8 GetHostInfo

GetHostInfo returns either a selected HostInfo record or all the HostInfo on the HiCommand™ Device Manager Server (see Table 3.47).

Table 3.47 GetHostInfo Command Parameters

Parameter Name	Required	Description
objectid	Optional (omit to return all HostInfo records)	Object ID of the HostInfo record to retrieve.

3.6 Server Management Commands

The Server Management commands provide some management support for the HiCommand™ Device Manager Server.

3.6.1 GetServerInfo

GetServerInfo returns information about the HiCommand™ Device Manager Server including Server version, Server URI, etc. There are no parameters.

3.6.2 GetLogFile

GetLogFile returns the requested HiCommand™ Device Manager Server log file (see Table 3.48).

Table 3.48 GetLogFile Command Parameters

Parameter Name	Required	Description
filename	Required	Name of the requested log file (access.log, service.log, error.log, or trace.log).

3.6.3 ModifyDebugLevel

ModifyDebugLevel sets the amount of debugging information generated by the HiCommand™ Device Manager Server (see Table 3.49). The debug level can range from 0 to 4 (0=Full Debugging trace, 1=Basic Information, 2=Warning, 3=Error, 4=Fatal).

Modifying the debug level affects the amount of information written to the `trace.log` file, subsequent to the change. The command does not affect Server configuration files, so any debug level modifications via this command does not affect the debug level when the Server is restarted.

Table 3.49 ModifyDebugLevel Command Parameters

Parameter Name	Required	Description
level	Required	New debug level (number between 0 and 4: 0=Full Debugging trace, 1=Basic Information, 2=Warning, 3=Error, 4=Fatal).

3.6.4 GetDebugLevel

GetDebugLevel returns the current debug level setting of the HiCommand™ Device Manager Server. The debug level affects the amount of information written to the **trace.log** file. There are no parameters.

3.6.5 GetAlerts

GetAlerts returns the alert messages previously generated by the HiCommand™ Device Manager Server (see Table 3.50). The returned alerts can be limited by specifying a time and/or a number of messages.

Table 3.50 GetAlerts Command Parameters

Parameter Name	Required	Description
countfilter	Optional (returned alerts not limited by total count, if omitted)	Maximum number of messages to return.
timefilter	Optional (returned alerts not limited by total count, if omitted)	Request only messages newer than the date and time. Required format is YYYY/MM/DD HH:MM:SS.

3.6.6 DeleteAlerts

DeleteAlerts deletes one or more alerts present on the HiCommand™ Device Manager Server (see Table 3.51). You can either delete a specific alert by specifying the alert number, or delete all alerts originating from a specific source by specifying the source.

Table 3.51 DeleteAlerts Command Parameters

Parameter Name	Required	Description
alertnum	Optional (either this parameter or the source parameter must be specified, but not both).	The number identifying the alert to delete.
source	Optional (either this parameter or the alertnumber parameter must be specified, but not both).	Identifies a source of alerts; all alerts from this source are deleted.

3.6.7 AddURLLink

AddURLLink adds a URL associated with an application and links it with a HiCommand™ object (see Table 3.52). If the HiCommand™ object already has an URLLink associated with it, the existing reference is overwritten and no error is returned.

Table 3.52 AddURLLink Command Parameters

Parameter Name	Required	Description
url	Required	The complete URL is required to launch the application or web page.
name	Required	Name of the application.
linkedid	Required	The object ID to link to. Must be a valid, existing HiCommand objectID.

3.6.8 DeleteURLLink

DeleteURLLink removes the association of an application or web page with an object in HiCommand™ Device Manager Server (see Table 3.53).

Table 3.53 DeleteURLLink Command Parameters

Parameter Name	Required	Description
objectid	Optional (Either objectid or linkedid must be specified).	The objectID of the URLLink to remove.
linkedid	Optional (Either objectid or linkedid must be specified).	The object ID of the linked object. All links to this object are removed.
description	Optional	Description of URLLink version information.

3.6.9 GetURLLink

GetURLLink gets any or all URLLink objects in the HiCommand™ Device Manager Server (see Table 3.54).

Table 3.54 GetURLLink Command Parameters

Parameter Name	Required	Description
objectid	Optional (Can be omitted when using the linkedid parameter to return all URLLink objects; do not use the linkedid parameter with the objectid parameter)	The objectID of the URLLink object to return.
linkedid	Optional (Can be omitted when using the objectid parameter to return all URLLink objects; do not use the linkedid parameter with the objectid parameter)	The object ID of the linked object. All links to this object are returned.

Chapter 4 HiCommand™ Device Manager Properties File

4.1 Using the Properties File to Specify Options

A properties file defines some characteristics of the application and provides default values for options and parameters. Any option that can be set from the command line can also be specified from the properties file.

- To specify an option, use the long option name as the key (for example, `--messagetrace` instead of `-t`), and use the option value as the property value.
- If the application output consistently is sent to an output file, instead of entering the following at the command line, you can set the property corresponding to the output option as **output=redirect.out**.

```
C:\HiCommand> HiCommandCLI http://localhost:1963/service  
GetStorageArray --output redirect.out serialnum=30051 model=HDS9960
```

Note: If the HiCommand™ Device Manager server's security level is set to the default (Basic Authentication) level, the User and Password options must be supplied either from the command line or from the property file.

- Whenever an option is specified both from the command line and from the properties file, the value from the command line is used. To specify options that have no parameters from the properties file, set the option to **true**, for example, `messagetrace=true`.
- Options that do not take arguments cannot be overridden from the command line. These options are specified in the properties file.

For more information on HiCommand™ Device Manager properties, please refer to the *HiCommand™ Server Installation and Configuration Guide* (MK-91HC002).

4.2 Using the Properties File to Specify Parameters

Any parameter set from the command line can also be specified from the properties file. To specify a parameter, use the parameter name as the key and the parameter value as the property value. For example, if application commands that require that a storage array model consistently require the same value, instead of entering the following at the command line, you can set the property corresponding to the model parameter as **model=HDS9960**.

```
C:\HiCommand> HiCommandCLI http://localhost:1963/service  
GetStorageArray serialnum=30057 model=HDS9960
```

Whenever a parameter is specified both from the command line and from the properties file, the value from the command line is used.

Note: Because many parameters are optional, specifying parameters from the properties file could cause inadvertent output. There is no way to override a parameter specified in the command-line in a manner that indicates that the parameter is not present.

4.3 Changing Specific Properties from the Command Line

4.3.1 Diagnostic Logging

The diagnostic log is a tool for the developers, which provides additional information about the program execution. The default file for the log is `HiCommandCLI.log`, located in the executing directory. The property file can be configured to use an alternate file, the `HiCommandCLI.logfile`. For example, specifying `HiCommandCLI.logfile` is the same as specifying `C:/Temp/cli.log`.

Remember to specify file names correctly in a Windows® environment. For example, controlling the level of diagnostic information.

- Debugging
`HiCommandCLI.loglevel=DEBUG`
- Warnings and Errors
`HiCommandCLI.loglevel=WARN`

Allowable values for the `HiCommandCLI.loglevel` property are `DEBUG`, `INFO`, and `WARN`.

4.3.2 Message Trace Output

The recording of the Request Messages sent to the HiCommand™ Device Manager Server and the received responses can be controlled using the `-t` or `--messagetrace` option. Like all options, the value can also be set in the properties file.

The file in which the messages are recorded can be specified in the properties file. The default file is in the executing directory, `MessageTrace.log`.

For example, the `HiCommandCLI.tracefile` property can be specified as
`HiCommandCLI.tracefile=C:/HiCommand/traffic.log`.

For more information on HiCommand™ Device Manager properties, please refer to the *HiCommand™ Server Installation and Configuration Guide (MK-91HC002)*.

4.3.3 Specifying the URL

Because the same HiCommand™ Device Manager Server may be used repeatedly, the URL can be specified in the properties file. The `HiCommandCLI.serverurl` property can specify the URL, `HiCommandCLI.serverurl=http://localhost:2001/service`.

When the URL is defined in the properties file, it can be omitted from the command line. For example:

```
C:\HiCommand> HiCommandCLI GetStorageArray serialnum=30057 model=HDS9960.
```

When the URL is also specified in the command line, that URL is used instead of the URL from the properties file.

4.3.4 Inputting Requests from XML Documents

Rather than specify the command and parameters from the command line, the request can originate from a file. The file must contain a valid XML document in the correct form for a HiCommand™ Device Manager request. The command line option **-i** or **--input** specifies the input file. The other application options are still valid. However, any command or parameter specified from the command line or in the properties file is ignored, because that information is already in the XML document.

4.3.5 Using the Message Trace File

When message tracing is turned on, each execution of the application writes over the message tracing file with the request and response of that execution, so that file never has more than one request and one response.

The request and the response are identified by labels. Each label includes a timestamp. The request also includes the HTTP header values set by the application. The request is the same string that is streamed over HTTP to the Server. The response is the string read from the HiCommand™ Device Manager Server via HTTP, before any parsing. Therefore, even if the application cannot make sense of the response (or does not handle the response correctly) the message trace contains a record of the response.

Chapter 5 Command Examples and Results

This chapter contains command examples and results. For each example, both the user and password options have been set correctly in advance in the property file.

5.1 Storage Array Commands

5.1.1 AddStorageArray

- Command example:

```
HiCommandCLI AddStorageArray -o "D:\logs\9980V AddStorageArray.log"  
"ipaddress=172.16.45.1" "family=HDS9900V" "userid=root" "arraypasswd=hicommand"  
"displayfamily=SANRISE9900V"
```

- Command result:

```
objectID=ARRAY.HDS9980V.0001  
name=SANRISE9980V@172.16.45.1  
description=SANRISE9980V (0001) at 172.16.45.1  
serialNumber=0001  
arrayFamily=HDS9900V  
arrayType=HDS9980V  
microcodeVersion=21-03-00/00  
agentVersion=02-05-14  
productName=RAID450  
controllerVersion=21-04-00-00/00  
numberOfControllers=4  
capacityInGB=4,018  
cacheInMB=-1  
sharedMemoryInMB=-1  
numberOfSpareDrives=-1  
freeCapacityInGB=3,585  
allocatedCapacityInGB=434  
hihsmCapacityInGB=0  
onDemandCapacityInGB=0  
totalFreeSpaceInGB=45  
largestFreeSpaceInGB=5  
capacityInKB=4,213,640,160  
freeCapacityInKB=3,759,001,920  
allocatedCapacityInKB=454,638,240  
hihsmCapacityInKB=0  
onDemandCapacityInKB=0  
totalFreeSpaceInKB=47,008,512  
largestFreeSpaceInKB=5,753,088  
multipathSupport=1  
securityStatus=2  
sequenceNumber=0001  
displayArrayFamily=SANRISE9900V  
displayArrayType=SANRISE9980V  
numberOfLUs=1,744  
numberOfAllocatedLUs=180  
numberOfUnallocatedLUs=1,564  
List of 1 CommParameters elements:  
  An instance of CommParameters  
  userID=root  
  userPassword=hicommand  
  ipAddress=172.16.45.1
```

5.1.2 GetStorageArray

5.1.2.1 subtarget=Path

- Command example 1 (without parameter objectid):

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_Path.log"  
"subtarget=Path" "model=HDS9980V" "serialnum=0001"  
"pathsubinfo=WWN,WWNGroup,HostInfo"
```

- Command result:

```
.  
  (Attributes of StorageArray are omitted here)  
.  
List of 554 Path elements:  
  An instance of Path  
    objectID=PATH.HDS9980V.0001.0.0.0  
    devNum=0  
    portID=0  
    domainID=0  
    scsiID=15  
    LUN=0  
    wwnSecurityValidity=true  
.  
  (repeated for other Path instances)  
.  
  An instance of Path  
    objectID=PATH.HDS9980V.0001.7.0.99  
    devNum=99  
    portID=7  
    domainID=0  
    scsiID=15  
    LUN=99  
    wwnSecurityValidity=true  
List of 256 WWN elements:  
  An instance of WWN  
    WWN=88.77.77.77.77.77.00.01  
    nickname=Jane  
  An instance of WWN  
    WWN=88.77.77.77.77.77.00.02  
    nickname=Jone  
.  
  (repeated for other WWN instances)  
.
```

■ Command example 2 (with parameter objectid):

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_Path.log"  
"subtarget=Path" "model=HDS9980V" "serialnum=0001" "objectid=  
PATH.HDS9980V.0001.7.0.99" "pathsubinfo=WWN,WWNGroup,HostInfo"
```

■ Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  . (Attributes of StorageArray are omitted here)  
.  
List of 1 Path elements:  
  An instance of Path  
    objectID=PATH.HDS9980V.0001.7.0.99  
    devNum=99  
    portID=7  
    domainID=0  
    scsiID=15  
    LUN=99  
    wwnSecurityValidity=true  
  List of 2 WWN elements:  
    An instance of WWN  
      WWN=88.77.77.77.77.77.00.01  
      nickname=Jane  
    An instance of WWN  
      WWN=88.77.77.77.77.77.00.02  
      nickname=Jone
```

5.1.2.2 subtarget=Port

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_Port.log"  
"subtarget=Port" "model=HDS9980V" "serialnum=0001" "port=7"  
"portsubinfo=WWN,WWNGroup,LUNGroup,HostStorageDomain"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  . (Attributes of StorageArray are omitted here)  
.  
List of 16 Port elements:  
  An instance of Port  
    objectID=PORT.HDS9980V.0001.0  
    serialNumber=0001  
    arrayType=HDS9980V  
    portID=0  
    portType=Fibre  
    fibreAddress=EF  
    topology=Fabric(off), FC-AL  
    displayName=CL1-A  
    lunSecurityEnabled=true  
    controllerID=1  
    worldWidePortName=50.06.0E.80.03.3A.99.00  
    channelSpeed=1  
  List of 2 WWN elements:  
    An instance of WWN  
      WWN=00.00.00.00.00.00.00.01  
      nickname=Jane  
    An instance of WWN  
      WWN=00.00.00.00.00.00.00.50  
      nickname=Jone  
  List of 2 HostStorageDomain elements:  
    An instance of HostStorageDomain  
      objectID=HSDOMAIN.HDS9980V.0001.0.0  
      portID=0  
      domainID=0  
      hostMode=Normal  
      displayName=CL1-A-0  
      nickname=1A-G00  
    An instance of HostStorageDomain  
      objectID=HSDOMAIN.HDS9980V.0001.0.1  
      portID=0  
      domainID=1  
      hostMode=Normal  
      displayName=CL1-A-1  
      nickname=HCMD0000  
  List of 1 WWNGroup elements:  
    An instance of WWNGroup  
      objectID= WWNGROUP.HDS9980V.0001.0.0.nickname  
      name=wwn group  
      nickname=nickname  
  List of 1 LUNGroup elements:  
    An instance of LUNGroup  
      objectID= LUNGROUP.HDS9980V.0001.0.0.hitachi  
      name=hitachi  
      nickname=hitachi  
  ... (repeated for other Port instances)
```

5.1.2.3 subtarget= LogicalUnit

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_LogicalUnit.log"
"subtarget=LogicalUnit" "model=HDS9980V" serialnum=0001" "lufilter=all"
"lusubinfo=Path,LDEV" "pathsubinfo=wwn,wwngroup,hostinfo"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1752 Lu elements:
  An instance of LogicalUnit
    objectID=LU.HDS9980V.0001.100
    devNum=100
    emulation=OPEN-3
    devCount=1
    devType=
    capacityInKB=2,403,360
    path=true
    commandDevice=false
    chassis=1
    arrayGroup=16
    raidType=RAID5(3D+1P)
    currentPortController=-1
    defaultPortController=-1
    isComposite=0
    sysVolFlag=0
    List of 3 Path elements:
      An instance of Path
      .
      . (Attributes of Path are omitted here)
      .
      An instance of Path
      .
      . (Attributes of Path are omitted here)
      .
      List of 2 WWN elements:
        An instance of WWN
          wwn=88.77.77.77.77.77.00.01
          nickname=Jane
        An instance of WWN
          wwn=88.77.77.77.77.77.3F.04
          nickname=Jone
      .
    . (repeated for other Path instances)
    .
    List of 1 Ldev elements:
      An instance of LDEV
        objectID=LDEV.HDS9980V.0001.100
        devNum=100
        emulation=OPEN-3
        cylinders=0
        isComposite=0
        sizeInKB=2,403,360
```

Results continue on the next page.

Continued from the previous page.

```
lba=1,201,680
raidType=RAID5(3D+1P)
slotSizeInKB=48
chassis=1
arrayGroup=16
path=true
onDemandDevice=false
devType=
isStandardLDEV=true
.
. (repeated for other LogicalUnit instances)
.
```

5.1.2.4 subtarget= LDEV

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_LDEV.log"
"subtarget=LDEV" "model=HDS9980V" "serialnum=0001" "devnum=0"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1756 Ldev elements:
An instance of LDEV
objectID=LDEV.HDS9980V.0001.0
devNum=0
emulation=OPEN-3
cylinders=0
isComposite=0
sizeInKB=2,403,360
lba=1,201,680
raidType=RAID5(3D+1P)
slotSizeInKB=48
chassis=1
arrayGroup=0
path=true
onDemandDevice=false
devType=
isStandardLDEV=true
.
. (repeated for other LDEV instances)
.
```

5.1.2.5 subtarget=PDEV

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_PDEV.log"  
"subtarget=PDEV" "model=HDS9980V" "serialnum=0001" "pdevid=1"  
"arraygroupobjid=ARRAYGROUP.HDS9980V.0001.0.0"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  . (Attributes of StorageArray are omitted here)  
.  
List of 80 PDEV elements:  
  An instance of PDEV  
    objectID=PDEV.HDS9980V.0001.0  
    chassis=1  
    arrayGroup=0  
    capacityInKB=72,000,000  
    row=-1  
    column=-1  
    depth=-1  
    role=  
    vendor=  
    model=DKR2D-J072FC  
    firmwareVersion=  
    serialNumber=  
    dkuType=HDS9900V  
    rpm=-1  
.  
  . (repeated for other PDEV instances)  
.
```

5.1.2.6 subtarget=ArrayGroup

- Command example 1:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_ArrayGroup.log"  
"subtarget=ArrayGroup" "model=HDS9980V" "serialnum=0001"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  . (Attributes of StorageArray are omitted here)  
.  
List of 20 ArrayGroup elements:  
  An instance of ArrayGroup  
    objectID=ARRAYGROUP.HDS9980V.0001.1.0  
    chassis=1  
    number=0  
    displayName=1-1-1  
    raidType=RAID5(3D+1P)  
    emulation=OPEN-3  
    diskType=DKR2D-J072FC  
    diskSize=72  
    controllerID=1  
    totalCapacity=203,925,600  
    allocatedCapacity=203,925,600  
    freeCapacity=0  
    hiHsmCapacity=0  
    onDemandCapacity=0  
    totalFreeSpace=9,126,144  
    largestFreeSpace=9,126,144  
.  
  . (repeated for other ArrayGroup instances)  
.
```

■ Command example 2:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_ArrayGroup.log"
"subtarget=ArrayGroup" "model=HDS9980V" "serialnum=0001" "objectID=
ARRAYGROUP.HDS9980V.0001.1.0"
```

■ Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1 ArrayGroup elements:
An instance of ArrayGroup
objectID= ARRAYGROUP.HDS9980V.0001.1.0
chassis=1
number=0
displayName=1-1-1
raidType=RAID5(3D+1P)
emulation=OPEN-3
diskType=DKR2D-J072FC
diskSize=72
controllerID=1
totalCapacity=203,925,600
allocatedCapacity=203,925,600
freeCapacity=0
hiHsmCapacity=0
onDemandCapacity=0
totalFreeSpace=9,126,144
largestFreeSpace=9,126,144
```

5.1.2.7 subtarget=PortController

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_PortController.log"
"subtarget=PortController" "model=HDS9980V" "serialnum=0001"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 4 PortController elements:
An instance of PortController
  objectID=CONTROLLER.HDS9980V.0001.1
  cluster=1
  card=1
  controllerID=1
  displayName=CHA-1P
  mode=1
  type=12
.
. (repeated for other PortController instances)
.
```

- Command example 2:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_PortController.log"
"subtarget=PortController" "model=HDS9980V" "serialnum=0001" "controllernum=1"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1 PortController elements:
An instance of PortController
  objectID=CONTROLLER.HDS9980V.0001.1
  cluster=1
  card=1
  controllerID=1
  displayName=CHA-1P
  mode=1
  type=12
```

5.1.2.8 subtarget=FreeSpace

- Command example 1:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_FreeSpace.log"  
"subtarget=FreeSpace" "model=HDS9980V" "serialnum=0001"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  . (Attributes of StorageArray are omitted here)  
.  
List of 20 ArrayGroup elements:  
  An instance of ArrayGroup  
    objectID=ARRAYGROUP.HDS9980V.0001.1.0  
    chassis=1  
    number=0  
    displayName=1-1-1  
    raidType=RAID5(3D+1P)  
    emulation=OPEN-3  
    diskType=DKR2D-J072FC  
    diskSize=72  
    controllerID=1  
    totalCapacity=203,925,600  
    allocatedCapacity=203,925,600  
    freeCapacity=0  
    hiHsmCapacity=0  
    onDemandCapacity=0  
    totalFreeSpace=9,126,144  
    largestFreeSpace=9,126,144  
    List of 1 FreeSpace elements:  
      An instance of FreeSpace  
        objectID=FREESPACE.HDS9980V.0001.1.0.85  
        sizeInKB=9,126,144  
        cylinders=0  
        fsControlIndex=85  
    .  
  . (repeated for other ArrayGroup instances)  
.
```

■ Command example 2:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_FreeSpace.log"  
"subtarget=FreeSpace" "model=HDS9980V" "serialnum=0001"  
"arraygroupobjid=ARRAYGROUP.HDS9980V.0001.1.16"
```

■ Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  . (Attributes of StorageArray are omitted here)  
.  
List of 1 ArrayGroup elements:  
  An instance of ArrayGroup  
    objectID=ARRAYGROUP.HDS9980V.0001.1.0  
    chassis=1  
    number=0  
    displayName=1-1-1  
    raidType=RAID5(3D+1P)  
    emulation=OPEN-3  
    diskType=DKR2D-J072FC  
    diskSize=72  
    controllerID=1  
    totalCapacity=203,925,600  
    allocatedCapacity=203,925,600  
    freeCapacity=0  
    hiHsmCapacity=0  
    onDemandCapacity=0  
    totalFreeSpace=9,126,144  
    largestFreeSpace=9,126,144  
  List of 1 FreeSpace elements:  
    An instance of FreeSpace  
      objectID=FREESPACE.HDS9980V.0001.1.0.85  
      sizeInKB=9,126,144  
      cylinders=0  
      fsControlIndex=85
```

5.1.2.9 subtarget=Filter

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_Filter.log"  
"subtarget=Filter" "model=HDS9980V" "serialnum=0001" "objectID=LU.HDS9980V.0001.100"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  . (Attributes of StorageArray are omitted here)  
.  
List of 1 Lu elements:  
  An instance of LogicalUnit  
    objectID=LU.HDS9980V.0001.100  
    devNum=100  
    emulation=OPEN-3  
    devCount=1  
    devType=  
    capacityInKB=2,403,360  
    path=true  
    commandDevice=false  
    chassis=1  
    arrayGroup=16  
    raidType=RAID5(3D+1P)  
    currentPortController=-1  
    defaultPortController=-1  
    isComposite=0  
    sysVolFlag=0
```

5.1.2.10 subtarget=HostStorageDomain

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_HostStorageDomain.log"
"subtarget=HostStorageDomain" "model=HDS9980V" "serialnum=0001" "port=23" "domain=1"
"hsdsubinfo=path,wwn,freeLun"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 41 HostStorageDomain elements:
  An instance of HostStorageDomain
    objectID=HSDOMAIN.HDS9980V.0001.0.0
    portID=0
    domainID=0
    hostMode=Normal
    displayName=CL1-A-0
    nickname=1A-G00
    List of 256 wwn elements:
      An instance of wwn
        wwn=88.77.77.77.77.77.00.01
        nickname=Jane
      An instance of wwn
        wwn=88.77.77.77.77.77.00.02
        nickname=Jone
    .
    . (repeated for other wwn instances)
    .
      List of 7 FreeLUN elements:
        An instance of FreeLUN
          lun=9
        An instance of FreeLUN
          lun=10
    .
    . (repeated for other FreeLUN instances)
    .
      List of 249 Path elements:
        An instance of Path
          objectID=PATH.HDS9980V.0001.7.0.10
          devNum=10
          portID=7
          domainID=0
          scsiID=15
          LUN=10
          wwnSecurityValidity=true
    .
    . (repeated for other Path instances)
    .
      An instance of HostStorageDomain
    .
    . (repeated for other HostStorageDomain instances)
    .
```

5.1.2.11 subtarget=Component

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9960 GetStorageArray_Component.log"  
"subtarget=Component" "model=HDS9960" "serialnum=10011"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
  objectID=ARRAY.HDS9960.10011  
  name=SANRISE2800@10.208.114.140  
  description=SANRISE2800 (10011) at 10.208.114.140  
  serialNumber=10011  
  arrayFamily=HDS9900  
  arrayType=HDS9960  
  microcodeVersion=01-13-56/00  
  agentVersion=02-01-06/00  
  productName=RAID400  
  controllerVersion=01-18-01-00/00  
  numberOfControllers=1  
  capacityInGB=1,326  
  cacheInMB=7,168  
  sharedMemoryInMB=-1  
  numberOfSpareDrives=-1  
  freeCapacityInGB=701  
  allocatedCapacityInGB=624  
  hihsmCapacityInGB=0  
  onDemandCapacityInGB=0  
  totalFreeSpaceInGB=124  
  largestFreeSpaceInGB=63  
  capacityInKB=1,390,713,840  
  freeCapacityInKB=735,710,400  
  allocatedCapacityInKB=655,003,440  
  hihsmCapacityInKB=0  
  onDemandCapacityInKB=0  
  totalFreeSpaceInKB=130,084,560  
  largestFreeSpaceInKB=67,096,800  
  multipathSupport=1  
  securityStatus=1  
  sequenceNumber=10011  
  displayArrayFamily=SANRISE2000  
  displayArrayType=SANRISE2800  
  numberOfLUs=626  
  numberOfAllocatedLUs=210  
  numberOfUnallocatedLUs=416  
List of 12 Component elements:  
  An instance of Component  
    name=DKC Battery  
    value=1  
    description=Normal  
  An instance of Component  
    name=DKC Cache  
    value=1  
    description=Normal  
  An instance of Component  
    name=DKC Cache Switch  
    value=1  
    description=Normal
```

Results continue on the next page.

Continued from the previous page.

```
An instance of Component
  name=DKC Environment
  value=1
  description=Normal
An instance of Component
  name=DKC Fan
  value=1
  description=Normal
An instance of Component
  name=DKC Power Supply
  value=1
  description=Normal
An instance of Component
  name=DKC Processor
  value=1
  description=Normal
An instance of Component
  name=DKC Shared Memory
  value=1
  description=Normal
An instance of Component
  name=DKU Drive
  value=1
  description=Normal
An instance of Component
  name=DKU Environment
  value=1
  description=Normal
An instance of Component
  name=DKU Fan
  value=1
  description=Normal
An instance of Component
  name=DKU Power Supply
  value=1
```

5.1.2.12 description=Normalsubtarget=CommParameters

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_CommParameters.log"  
"subtarget=CommParameters" "model=HDS9980V" "serialnum=0001"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  . (Attributes of StorageArray are omitted here)  
.  
List of 1 CommParameters elements:  
  An instance of CommParameters  
    userID=root  
    userPassword=hicommand  
    ipAddress=172.16.45.1
```

5.1.2.13 subtarget=ReplicationInfo

- Command example:

```
HiCommandCLI GetStorageArray -o "D:\logs\9980V GetStorageArray_ReplicationInfo.log"
"subtarget=ReplicationInfo" "model=HDS9980V" "serialnum=0001"
"replicationsubinfo=LogicalUnit"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 2 ReplicationInfo elements:
An instance of ReplicationInfo
  objectID=REPINFO.0001.100.0001.101
  pvolSerialNumber=0001
  pvolArrayType=HDS9980V
  pvolDevNum=100
  pvolObjectID=LU.HDS9980V.0001.100
  svolSerialNumber=0001
  svolArrayType=HDS9980V
  svolDevNum=101
  svolObjectID=LU.HDS9980V.0001.101
  replicationFunction=ShadowImage
  fenceLevel=Data
  status=1
List of 2 Lu elements:
  An instance of LogicalUnit
    objectID=LU.HDS9980V.0001.100
    devNum=100
    emulation=OPEN-3
    devCount=1
    devType=
    capacityInKB=2,403,360
    path=true
    commandDevice=false
    chassis=1
    arrayGroup=16
    raidType=RAID5(3D+1P)
    currentPortController=-1
    defaultPortController=-1
    isComposite=0
    sysVolFlag=0
  An instance of LogicalUnit
    objectID=LU.HDS9980V.0001.101
    devNum=101
    emulation=OPEN-3
    devCount=1
    devType=
    capacityInKB=2,403,360
    path=true
    commandDevice=false
    chassis=1
    arrayGroup=16
    raidType=RAID5(3D+1P)
    currentPortController=-1
    defaultPortController=-1
    isComposite=0
    sysVolFlag=0
... (repeated for other ReplicationInfo instances)
```

5.1.3 DeleteStorageArray

- Command example:

```
HiCommandCLI DeleteStorageArray -o "D:\logs\9980V DeleteStorageArray.log"  
"serialnum=0001" "model=HDS9980V"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

5.1.4 AddHostStorageDomain

- Command example:

```
HiCommandCLI AddHostStorageDomain -o "D:\logs\9980V AddHostStorageDomain.log"  
"serialnum=0001" "model=HDS9980V" "port=16" "domain=1"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  (Attributes of StorageArray are omitted here)  
.  
List of 1 HostStorageDomain elements:  
  An instance of HostStorageDomain  
    objectID=HSDOMAIN.HDS9980V.0001.16.1  
    portID=16  
    domainID=1  
    hostMode=Normal  
    displayName=CL2-A-1  
    nickname=HCMD1001
```

5.1.5 DeleteHostStorageDomain

- Command example:

```
HiCommandCLI DeleteHostStorageDomain -o "D:\logs\9980V DeleteHostStorageDomain.log"  
"serialnum=0001" "model=HDS9980V" "port=16" "domain=1" "deletionoption=lusekeep"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

5.1.6 AddArrayGroup

- Command example:

```
HiCommandCLI AddArrayGroup -o "D:\logs\AddArrayGroup.log" "serialnum=3001"  
"model=SUNT3B" "chassis=0" "raidtype=RAID0(2D)" "pdevnums=0,1"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  . (Attributes of StorageArray are omitted here)  
.  
List of 1 ArrayGroup elements:  
  An instance of ArrayGroup  
    objectID=ARRAYGROUP.SUNT3B.3001.0.0  
    chassis=0  
    number=0  
    displayName=u1ag1  
    raidType=RAID0(2D)  
    diskType=ST336605FSUN36G  
    diskSize=-0  
    controllerID=0  
    totalCapacity=0  
    allocatedCapacity=0  
    freeCapacity=0  
    hiHsmCapacity=0  
    onDemandCapacity=0  
    totalFreeSpace=-1  
    largestFreeSpace=-1  
  List of 2 PDEV elements:  
    An instance of PDEV  
      objectID=PDEV.SUNT3B.3001.0  
      chassis=0  
      arrayGroup=0  
      capacityInKB=-1  
      row=0  
      column=0  
      depth=1  
      role=data  
      vendor=SEAGATE  
      model=ST336605FSUN36G  
      firmwareVersion=A338  
      serialNumber=3FP1A6RZ  
      dkuType=SUNT3  
      rpm=-1  
    An instance of PDEV  
      objectID=PDEV.SUNT3B.3001.1  
      chassis=0  
      arrayGroup=0  
      capacityInKB=-1  
      row=0  
      column=1  
      depth=1  
      role=data  
      vendor=SEAGATE  
      model=ST336605FSUN36G  
      firmwareVersion=A338  
      serialNumber=3FP19XNW  
      dkuType=SUNT3  
      rpm=-1
```

5.1.7 DeleteArrayGroup

- Command example:

```
HiCommandCLI DeleteArrayGroup -o "D:\logs\T3B DeleteArrayGroup.log" "serialnum=3001"
"model=SUNT3B" "chassis=0" "groupnum=2"
```

- Command result:

```
RESPONSE:
(Command completed; empty list returned)
```

5.1.8 AddSpareDrive

- Command example:

```
HiCommandCLI AddSpareDrive -o "D:\logs\9200 AddSpareDrive.log" "serialnum=0001"
"model=HDS9200" "pdevid=5"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1 PDEV elements:
  An instance of PDEV
    objectID=PDEV.HDS9200.0001.5
    chassis=0
    arrayGroup=-1
    capacityInKB=108,003,328
    row=0
    column=5
    depth=1
    role=spareuninitialized
    vendor=SEAGATE
    model=ST39103FC
    firmwareVersion=FF78
    serialNumber=
    dkuType=HDS9200
    rpm=0
```

5.1.9 DeleteSpareDrive

- Command example:

```
HiCommandCLI DeleteSpareDrive -o "D:\logs\9200 DeleteSpareDrive.log" "serialnum=0001"
"model=HDS9200" "pdevid=5"
```

- Command result:

```
RESPONSE:
(Command completed; empty list returned)
```

5.1.10 AddLogicalUnit

- Command example 1:

```
HiCommandCLI AddLogicalUnit -o "D:\logs\9980V AddLogicalUnit.log" "serialnum=0001"
"model=HDS9980V" "chassis=4" "groupnum=32" "capacity=1000000" "emulation=OPEN-3"
"lusubinfo=LDEV"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1 Lu elements:
  An instance of LogicalUnit
    objectID=LU.HDS9980V.0001.176
    devNum=176
    emulation=OPEN-3
    devCount=1
    devType=
    capacityInKB=1,000,080
    path=false
    commandDevice=false
    chassis=4
    arrayGroup=32
    raidType=RAID5(3D+1P)
    currentPortController=-1
    defaultPortController=-1
    isComposite=0
    trueCopyVolumeType=Unknown
    shadowImageVolumeType=Unknown
  List of 1 Ldev elements:
    An instance of LDEV
      objectID=LDEV.HDS9980V.0001.176
      devNum=176
      emulation=OPEN-3
      cylinders=0
      isComposite=0
      sizeInKB=1,000,080
      lba=14,351,040
      raidType=RAID5(3D+1P)
      slotSizeInKB=48
      chassis=4
      arrayGroup=32
      path=false
      onDemandDevice=false
      devType=
      isStandardLDEV=false
```

■ Command example 2:

```
HiCommandCLI AddLogicalUnit -o "D:\logs\9200 AddLogicalUnit.log" "serialnum=0001"
"model=HDS9200" "chassis=0" "groupnum=2" "capacity=1000000" "defportctl=0"
"lusubinfo=LDEV"
```

■ Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1 Lu elements:
  An instance of LogicalUnit
    objectID=LU.HDS9200.0001.58
    devNum=58
    devCount=1
    devType=
    capacityInKB=1,000,000
    path=false
    commandDevice=false
    chassis=0
    arrayGroup=2
    raidType=RAID1(1D+1P)
    currentPortController=0
    defaultPortController=0
    isComposite=0
    trueCopyVolumeType=Unknown
    shadowImageVolumeType=Unknown
  List of 1 Ldev elements:
    An instance of LDEV
      objectID=LDEV.HDS9200.0001.58
      devNum=58
      cylinders=-1
      isComposite=0
      sizeInKB=1,000,000
      lba=-1
      raidType=RAID1(1D+1P)
      slotSizeInKB=-1
      chassis=0
      arrayGroup=2
      path=false
      onDemandDevice=false
      devType=
      isStandardLDEV=true
```

5.1.11 DeleteLogicalUnit

- Command example:

```
HiCommandCLI DeleteLogicalUnit -o "D:\logs\ SUNT300 DeleteLogicalUnit.log"  
"serialnum=3000" "model=SUNT300" "devnum=1"
```

- Command result:

```
RESPONSE:  
(Command completed; empty list returned)
```

5.1.12 ModifyLogicalUnit

- Command example:

```
HiCommandCLI ModifyLogicalUnit -o "D:\logs\9980V ModifyLogicalUnit.log"  
"serialnum=0001" "model=HDS9980V" "devnum=1" "commanddevice=true"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
.(Attributes of StorageArray are omitted here)  
.  
List of 1 Lu elements:  
  An instance of LogicalUnit  
    objectID=LU.HDS9980V.0001.1  
    devNum=1  
    emulation=OPEN-3  
    devCount=1  
    devType=CommandDevice  
    capacityInKB=2,403,360  
    path=true  
    commandDevice=true  
    chassis=1  
    arrayGroup=0  
    raidType=RAID5(3D+1P)  
    currentPortController=-1  
    defaultPortController=-1  
    isComposite=0  
    trueCopyVolumeType=Unknown  
    shadowImageVolumeType=Unknown  
    sysVolFlag=0
```

5.1.13 AddLUSE

- Command example:

```
HiCommandCLI AddLUSE -o "D:\logs\9980V AddLUSE.log" "serialnum=15001"
"model=HDS9980V" "devnums=1001,1002"
```

- Command result:

```
RESPONSE:
An instance of StorageArray

.
. (Attributes of StorageArray are omitted here)
.
List of 1 Lu elements:
  An instance of LogicalUnit
    objectID=LU.HDS9980V.15001.1001
    devNum=1,001
    emulation=OPEN-3
    devCount=2
    devType=
    capacityInKB=4,806,720
    path=false
    commandDevice=false
    chassis=3
    arrayGroup=16
    raidType=RAID5(3D+1P)
    currentPortController=-1
    defaultPortController=-1
    isComposite=1
    trueCopyVolumeType=Simplex
    shadowImageVolumeType=Simplex
    sysVolFlag=0
  List of 2 Ldev elements:
    An instance of LDEV
      objectID=LDEV.HDS9980V.15001.1001
      devNum=1,001
      emulation=OPEN-3
      cylinders=0
      isComposite=1
      sizeInKB=2,403,360
      lba=1,201,680
      raidType=RAID5(3D+1P)
      slotSizeInKB=48
      chassis=3
      arrayGroup=16
      path=false
      onDemandDevice=false
      devType=
      isStandardLDEV=true
    An instance of LDEV
      objectID=LDEV.HDS9980V.15001.1002
      devNum=1,002
      emulation=OPEN-3
      cylinders=0
      isComposite=1
      sizeInKB=2,403,360
      lba=1,201,680
      raidType=RAID5(3D+1P)
      slotSizeInKB=48
```

Results continue on the next page.

Continued from the previous page.

```
chassis=3
arrayGroup=16
path=false
onDemandDevice=false
devType=
isStandardLDEV=true
```

5.1.14 DeleteLUSE

- Command example:

```
HiCommandCLI DeleteLUSE -o "D:\ logs\9980V DeleteLUSE.log" "serialnum=0001"
"model=HDS9980V" "devnum=209"
```

- Command result:

```
RESPONSE:
(Command completed; empty list returned)
```

5.1.15 ModifyPort

- Command example:

```
HiCommandCLI ModifyPort -o "D:\logs\9980V ModifyPort.log" "serialnum=0001"
"model=HDS9980V" "port=16" "domain=0" "hostmode=Normal" "hostmode2=UA (06/2A00)
suppress Mode;HISUP Mode;ftServer Connection Mode 2;SRC Read Command Reject Mode"
"topo=Fabric(off), FC-AL" "fibreadr=73" "lunsec=false" "portoption=Reset ALL LIP
Port Mode;Reset/LIP Mode (Process)" "channelspeed=12"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
. (Attributes of StorageArray are omitted here)
.
List of 1 Port elements:
An instance of Port
objectID=PORT.HDS9980V.0001.16
serialNumber=0001
arrayType=HDS9980V
portID=16
portType=Fibre
fibreAddress=73
topology=Fabric(off), FC-AL
displayName=CL2-A
lunSecurityEnabled=false
controllerID=5
worldWidePortName=50.06.0E.80.03.3A.99.10
channelSpeed=12
List of 1 HostStorageDomain elements:
An instance of HostStorageDomain
objectID=HSDOMAIN.HDS9980V.0001.16.0
portID=16
domainID=0
hostMode=Normal
hostMode2=UA (06/2A00) suppress Mode;HISUP Mode;ftServer Connection Mode 2;
SRC Read Command Reject Mode
```

5.1.16 ModifyPortController

- Command example:

```
HiCommandCLI ModifyPortController -o "D:\logs\9980V ModifyPortController.log"  
"serialnum=0001" "model=HDS9980V" "ctrlid=1" "adaptermode=1"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
. (Attributes of StorageArray are omitted here)  
.  
List of 1 PortController elements:  
An instance of PortController  
objectID=CONTROLLER.HDS9980V.0001.1  
cluster=1  
card=1  
controllerID=1  
displayName=CHA-1P  
mode=1  
type=12
```

5.1.17 RefreshStorageArrays

- Command example:

```
HiCommandCLI RefreshStorageArrays -o "D:\logs\RefreshStorageArrays.log" "interval=1"
```

- Command result:

```
RESPONSE:  
[An instance of StorageArray  
.  
. (Attributes of StorageArray are omitted here)  
.  
List of 1 CommParameters elements:  
An instance of CommParameters  
userID=  
userPassword=  
ipAddress2=172.16.50.2  
ipAddress=172.16.50.1  
]  
[An instance of StorageArray  
.  
. (Attributes of StorageArray are omitted here)  
.  
List of 1 CommParameters elements:  
An instance of CommParameters  
snmpWrCommunity=public  
ipAddress=172.16.40.1  
snmpRdCommunity=public  
snmpPort=161  
]  
.  
. (repeated for other StorageArray instances)  
.
```

5.2 Logical Group Commands

5.2.1 AddLogicalGroup

- Command example 1:

```
HiCommandCLI AddLogicalGroup -o "D:\logs\AddLogicalGroup.log" "groupname=toro"
"iconfile=group12.gif"
```

- Command result:

```
RESPONSE:
An instance of LogicalGroup
  objectID=GROUP.2
  name=toro
  logicalPath=toro
  icon=group12.gif
  capacity=0
  percentUsed=0
```

- Command example 2:

```
HiCommandCLI AddLogicalGroup -o "D:\logs\AddLogicalGroup.log" "groupname=toro"
"iconfile=group12.gif" "parent=GROUP.0"
```

- Command result:

```
RESPONSE:
An instance of LogicalGroup
  objectID=GROUP.1
  name=toro
  logicalPath=root/toro
  parentID=GROUP.0
  icon=group12.gif
  capacity=0
  percentUsed=0
```

5.2.2 DeleteLogicalGroup

- Command example:

```
HiCommandCLI DeleteLogicalGroup -o "D:\logs\DeleteLogicalGroup.log"
"objectid=GROUP.0"
```

- Command result:

```
RESPONSE:
(Command completed; no data returned)
```

5.2.3 ModifyLogicalGroup

- Command example:

```
HiCommandCLI ModifyLogicalGroup -o "D:\logs\ModifyLogicalGroup.log"  
"objectID=GROUP.0" "groupname=snow" "iconfile=GROUP.13"
```

- Command result:

```
RESPONSE:  
An instance of LogicalGroup  
  objectID=GROUP.1  
  name=snow  
  logicalPath=root/snow  
  parentID=GROUP.0  
  icon=group12.gif  
  capacity=0  
  percentUsed=0
```

5.2.4 GetLogicalGroup

- Command example:

```
HiCommandCLI GetLogicalGroup -o "D:\logs\GetLogicalGroup.log"  
"subtarget=Host,HostStorageDomain,path"
```

- Command result:

```
RESPONSE:  
An instance of LogicalGroup  
  objectID=GROUP.1  
  name=toro  
  parentID=GROUP.0  
  icon=group12.gif  
  capacity=0  
  percentUsed=0  
List of 3 GroupElement elements:  
  An instance of Host  
    objectID=HOST.1  
    name=hit  
    ipAddress=192.168.32.63  
    capacityInKB=0  
  An instance of HostStorageDomain  
    objectID=HSDOMAIN.HDS9980V.0001.0.0  
    portID=0  
    domainID=0  
    hostMode=HP  
    displayName=CL1-A-0  
    nickname=1A-G00  
  An instance of Path  
    objectID=PATH.HDS9980V.0001.0.0.101  
    devNum=101  
    portID=0  
    domainID=0  
    scsiID=15  
    LUN=8  
    wwnSecurityValidity=true  
.  
.(repeated for other LogicalGroup instances)  
.
```

■ Command example 2:

```
HiCommandCLI GetLogicalGroup -o "D:\logs\GetLogicalGroup.log" "objectID=GROUP.1"
"subtarget=Host,HostStorageDomain,path"
```

■ Command result:

```
RESPONSE:
An instance of LogicalGroup
  objectID=GROUP.1
  name=toro
  parentID=GROUP.0
  icon=group12.gif
  capacity=0
  percentUsed=0
List of 3 GroupElement elements:
  An instance of Host
    objectID=HOST.1
    name=hit
    ipAddress=192.168.32.63
    capacityInKB=0
  An instance of HostStorageDomain
    objectID=HSDOMAIN.HDS9980V.0001.0.0
    portID=0
    domainID=0
    hostMode=HP
    displayName=CL1-A-0
    nickname=1A-G00
  An instance of Path
    objectID=PATH.HDS9980V.0001.0.0.101
    devNum=101
    portID=0
    domainID=0
    scsiID=15
    LUN=8
    wwnSecurityValidity=true
```

■ Command example 3:

```
HiCommandCLI GetLogicalGroup -o "D:\logs\GetLogicalGroup.log"
```

■ Command result:

```
RESPONSE:
An instance of LogicalGroup
  objectID=GROUP.1
  name=toro
  parentID=GROUP.0
  icon=group12.gif
  capacity=0
  percentUsed=0
.
. (repeated for other LogicalGroup instances)
.
```

5.2.5 AddObjectForLogicalGroup

- Command example:

```
HiCommandCLI AddObjectForLogicalGroup -o "D:\logs\9980V AddObjectForLogicalGroup.log"
"objectID=GROUP.0" "groupelements=PATH.HDS9980V.0001.0.0.1"
```

- Command result:

```
RESPONSE:
An instance of LogicalGroup
  objectID=GROUP.1
  name=toro
  parentID=GROUP.0
  icon=group12.gif
  capacity=0
  percentUsed=0
List of 1 GroupElement elements:
  An instance of Path
    objectID=PATH.HDS9980V.0001.0.0.0
    devNum=0
    portID=0
    domainID=0
    scsiID=15
    LUN=0
    wwnSecurityValidity=true
```

5.2.6 DeleteObjectForLogicalGroup

- Command example:

```
HiCommandCLI DeleteObjectForLogicalGroup -o "D:\logs\9980V
DeleteObjectForLogicalGroup.log" "objectID=GROUP.0"
"groupelements=PATH.HDS9980V.0001.0.0.1"
```

- Command result:

```
RESPONSE:
(Command completed; no data returned)
```

5.2.7 AddLunScan

- Command example:

```
HiCommandCLI AddLunScan -o "D:\logs\9980V AddLunScan.log" "serialnum=0001"
"model=HDS9980V"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 94 HostStorageDomain elements:
An instance of HostStorageDomain
  objectID=HSDOMAIN.HDS9980V.0001.0.0
  portID=0
  domainID=0
  hostMode=HP
  displayName=CL1-A-0
  nickname=1A-G00
.
. (repeated for other HostStorageDomain instances)
.
List of 283 Path elements:
An instance of Path
  objectID=PATH.HDS9980V.0001.0.0.101
  devNum=101
  portID=0
  domainID=0
  scsiID=15
  LUN=8
  wwnSecurityValidity=true
.
. (repeated for other Path instances)
.
```

5.3 LUN Commands

5.3.1 AddLunGroup

- Command example:

```
HiCommandCLI AddLunGroup -o "D:\ logs\9960 AddLunGroup.log" "serialnum=0001"  
"model=HDS9960" "port=19"  
"groupelements=PATH.HDS9960.0001.16.2.0,PATH.HDS9960.0001.16.2.1" "nickname=hitachi"  
"name=hitachi"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  (Attributes of StorageArray are omitted here)  
.  
List of 1 Port elements:  
  An instance of Port  
  .  
    (Attributes of Port are omitted here)  
  .  
List of 1 LUNGroup elements:  
  An instance of LUNGroup  
    objectID=LUNGROUP.HDS9960.0001.0.19.hitachi  
    name=hitachi  
    nickname=hitachi  
    List of 2 Path elements:  
      An instance of Path  
        objectID=PATH.HDS9960.0001.16.2.0  
        devNum=0  
        portID=19  
        domainID=16  
        scsiID=15  
        LUN=8  
        wwnSecurityValidity=true  
      .  
    . (repeated for other Path instances)  
  .
```

5.3.2 DeleteLunGroup

- Command example:

```
HiCommandCLI DeleteLunGroup -o "D:\logs\9960 DeleteLunGroup.log" "serialnum=10011"  
"model=HDS9960" "port=19" "Lungroupid=LUNGROUP.HDS9960.10011.19.LG00"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

5.3.3 ModifyLunGroup

- Command example:

```
HiCommandCLI ModifyLunGroup -o "D:\logs\9960 ModifyLunGroup.log" "serialnum=0001"  
"model=HDS9960" "port=19"  
"groupelements=PATH.HDS9960.0001.16.2.0,PATH.HDS9960.0001.16.2.1"  
"objectid=LUNGROUP.HDS9960.0001.19.hitachi" "nickname=hitachi" "name=hitachi"
```

- Command results:

```
RESPONSE:  
An instance of StorageArray  
.  
  (Attributes of StorageArray are omitted here)  
.  
List of 1 Port elements:  
  An instance of Port  
  .  
  (Attributes of Port are omitted here)  
  .  
List of 1 LUNGroup elements:  
  An instance of LUNGroup  
  objectID= LUNGROUP.HDS9960.0001.0.19.hitachi  
  name=hitachi  
  nickname=hitachi  
  List of 2 Path elements:  
  An instance of Path  
  objectID=PATH.HDS9960.0001.16.2.0  
  devNum=0  
  portID=19  
  domainID=16  
  scsiID=15  
  LUN=8  
  wwnSecurityValidity=true  
  .  
  (repeated for other Path instances)  
  .
```

5.3.4 AddWWNForLUNGroup

- Command example:

```
HiCommandCLI AddWWNForLUNGroup -o "D:\logs\9960 AddWWNForLUNGroup.log"  
"serialnum=0001" "model=HDS9960" "port=19"  
"objectid=LUNGROUP.HDS9960.0001.19.hitachi"  
"wwn=11.55.77.99.33.00.22.66,AA.AA.AA.AA.AA.AA.AA.AA"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  (Attributes of StorageArray are omitted here)  
.  
List of 1 Port elements:  
  An instance of Port  
  .  
  (Attributes of Port are omitted here)  
  .  
List of 1 LUNGroup elements:  
  An instance of LUNGroup  
    objectID= LUNGROUP.HDS9960.0001.0.19.hitachi  
    name=hitachi  
    nickname=hitachi  
    List of 2 wwn elements:  
      An instance of wwn  
        wwn=11.55.77.99.33.00.22.66  
  .  
  (repeated for other Path instances)  
  .
```

5.3.5 DeleteWWNForLUNGroup

- Command example:

```
HiCommandCLI DeletewWNForLUNGroup -o "D:\logs\9960 DeletewWNForLUNGroup.log"  
"serialnum=0001" "model=HDS9960" "port=19"  
"objectid=LUNGROUP.HDS9960.0001.19.hitachi"  
"wwn=11.55.77.99.33.00.22.66,AA.AA.AA.AA.AA.AA.AA.AA"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

5.3.6 AddLun

- Command example 1:

```
HiCommandCLI AddLun -o "D:\logs\9960 AddLun.log" "serialnum=0001" "model=HDS9960"
"name=liu" "port=3" "scsi=15" "lun=127" "lusedevnums=127,128"
"wwn=AA.AA.AA.AA.AA.AA.AA.AA"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
:
:
List of 1 Path elements:
  objectID=PATH.HDS9960.0001.3.16.127
  name=liu
  devNum=127
  portID=3
  domainID=16
  scsiID=15
  LUN=127
  wwnSecurityValidity=true
List of 1 WWN elements:
  An instance of WWN
    WWN=AA.AA.AA.AA.AA.AA.AA.AA
    nickname=Jane
List of 2 Ldev elements:
  An instance of LDEV
    objectID=LDEV.HDS9960.0001.127
    devNum=127
    emulation=OPEN-8
    cylinders=50
    isComposite=1
    sizeInKB=36,000
    lba=72,000
    raidType=RAID5(3D+1P)
    slotSizeInKB=48
    chassis=1
    arrayGroup=7
    path=true
    onDemandDevice=false
    devType=
    isStandardLDEV=false
  An instance of LDEV
    objectID=LDEV.HDS9960.0001.128
    devNum=128
    emulation=OPEN-8
    cylinders=50
    isComposite=1
    sizeInKB=36,000
    lba=72,000
    raidType=RAID5(3D+1P)
    slotSizeInKB=48
    chassis=1
    arrayGroup=7
    path=false
    onDemandDevice=false
    devType=
    isStandardLDEV=false
```

■ Command example 2:

```
HiCommandCLI AddLun -o "D:\logs\9980V AddLun.log" "serialnum=0001" "model=HDS9980V"
"name=hitachi" "port=1" "domain=4" "scsi=15" "lun=114" "lusedevnums=1001,1002"
```

■ Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1 Path elements:
  An instance of Path
    objectID=PATH.HDS9980V.0001.1.4.1001
    name=hitachi
    devNum=1,001
    portID=1
    domainID=4
    scsiID=15
    LUN=114
    wwnSecurityValidity=true
  List of 2 Ldev elements:
    An instance of LDEV
      objectID=LDEV.HDS9980V.0001.1001
      devNum=1,001
      emulation=OPEN-3
      cylinders=0
      isComposite=1
      sizeInKB=2,403,360
      lba=1,201,680
      raidType=RAID5(3D+1P)
      slotSizeInKB=48
      chassis=3
      arrayGroup=16
      path=true
      onDemandDevice=false
      devType=
      isStandardLDEV=true
    An instance of LDEV
      objectID=LDEV.HDS9980V.0001.1002
      devNum=1,002
      emulation=OPEN-3
      cylinders=0
      isComposite=1
      sizeInKB=2,403,360
      lba=1,201,680
      raidType=RAID5(3D+1P)
      slotSizeInKB=48
      chassis=3
      arrayGroup=16
      path=false
      onDemandDevice=false
      devType=
      isStandardLDEV=true
```

- Command example 3:

```
HiCommandCLI AddLun -o "D:\logs\9980V AddLun.log" "serialnum=0001" "model=HDS9980V"  
"name=hitachi" "port=1" "domain=4" "scsi=15" "lun=120" "devnum=1006"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  (Attributes of StorageArray are omitted here)  
.  
List of 1 Path elements:  
  An instance of Path  
    objectID=PATH.HDS9980V.0001.1.4.1006  
    name=hitachi  
    devNum=1,006  
    portID=1  
    domainID=4  
    scsiID=15  
    LUN=120  
    wwnSecurityValidity=true
```

5.3.7 DeleteLun

- Command example:

```
HiCommandCLI DeleteLun -o "D:\logs\9980V DeleteLun.log" "serialnum=0001"  
"model=HDS9980V" "port=1" "domain=1" "devnum=1" "deletionoption=lusekeep"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

5.3.8 AddWWNForHostStorageDomain

- Command example:

```
HiCommandCLI AddWWNForHostStorageDomain -o "D:\logs\9980V  
AddWWNForHostStorageDomain.log" "serialnum=0001" "model=HDS9980V" "port=1" "domain=1"  
"wwn=22.33.44.55.44.55.44.33,11.33.22.33.44.33.22.22"
```

- Command result:

```
RESPONSE:  
An instance of StorageArray  
.  
  (Attributes of StorageArray are omitted here)  
.  
List of 1 HostStorageDomain elements:  
  An instance of HostStorageDomain  
    objectID=HSDOMAIN.HDS9980V.0001.1.1  
    portID=1  
    domainID=1  
    hostMode=Normal  
    displayName=CL1-B-1  
    nickname=HCMD0103  
    List of 2 WWN elements:  
      An instance of WWN  
        WWN=22.33.44.55.44.55.44.33  
        nickname=Jane  
      An instance of WWN  
        WWN=11.33.22.33.44.33.22.22  
        nickname=Jone
```

5.3.9 DeleteWWNForHostStorageDomain

- Command example:

```
HiCommandCLI DeleteWWNForHostStorageDomain -o "D:\logs\9980V  
DeleteWWNForHostStorageDomain.log" "serialnum=0001" "model=HDS9980V" "port=1"  
"domain=1" "wwn=A.AA.AA.AA.AA.AA.AA.AA"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

5.3.10 DeleteWWN

- Command example:

```
HiCommandCLI DeleteWWN -o "D:\logs\9980V DeleteWWN.log" "serialnum=0001"  
"model=HDS9980V" "port=0" "wwn=11.22.33.22.44.55.33.11"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

5.3.11 AddWWNGroup

- Command example:

```
HiCommandCLI AddWWNGroup -o "D:\logs\9960 AddWWNGroup.log" "serialnum=0001"
"model=HDS9960" "port=19" "wwn=AA.AA.AA.AA.AA.AA.AA.AA" "nickname=hitachi"
"name=hitachi"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1 Port elements:
An instance of Port
.
. (Attributes of Port are omitted here)
.
List of 1 WWNGroup elements:
An instance of WWNGroup
  objectID=WWNGROUP.HDS9960.0001.0.19.hitachi
  name=hitachi
  nickname=hitachi
  List of 2 WWN elements:
  An instance of WWN
    WWN= 11.55.77.99.33.00.22.66
    nickname=Jane
  An instance of WWN
    WWN= AA.AA.AA.AA.AA.AA.AA.AA
    nickname=Jone
```

5.3.12 ModifyWWNGroup

- Command example:

```
HiCommandCLI ModifyWWNGroup -o "D:\logs\9960 ModifyWWNGroup.log" "serialnum=0001"
"model=HDS9960" "port=19" "objectid=WWNGROUP.HDS9960.0001.19.hitachi"
"wwn=AA.AA.AA.AA.AA.AA.AA.AA, BB.BB.BB.BB.BB.BB.BB.BB" "nickname=hitachi"
"name=hitachi"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1 Port elements:
An instance of Port
.
. (Attributes of Port are omitted here)
.
List of 1 WWNGroup elements:
An instance of WWNGroup
  objectID=WWNGROUP.HDS9960.0001.0.19.hitachi
  name=hitachi
  nickname=hitachi
  List of 2 WWN elements:
  An instance of WWN
    WWN= AA.AA.AA.AA.AA.AA.AA.AA
    nickname=Jane
  An instance of WWN
    WWN= BB.BB.BB.BB.BB.BB.BB.BB
    nickname=Jone
```

5.3.13 DeleteWWNGroup

- Command example:

```
HiCommandCLI DeleteWwnGroup -o "D:\\logs\\9960 DeleteWwnGroup.log" "serialnum=10011"
"model=HDS9960" "port=19" "wwngroup=wg"
```

- Command result:

```
RESPONSE:
(Command completed; no data returned)
```

5.3.14 AddWWNForLun

- Command example:

```
HiCommandCLI AddWwnForLun -o "D:/logs/9200 AddWwnForLun.log" "serialnum=0001"
"model=HDS9200" "port=16" "devnum=16" "domain=0" "wwn= AA.AA.AA.AA.AA.AA.AA.AA"
```

- Command result:

```
RESPONSE:
An instance of StorageArray
.
. (Attributes of StorageArray are omitted here)
.
List of 1 Path elements:
An instance of Path
objectID=PATH.HDS9200.0001.16.0.16
devNum=16
portID=16
domainID=0
scsiID=15
LUN=16
wwnSecurityValidity=true
List of 1 WWN elements:
An instance of WWN
WWN=AA.AA.AA.AA.AA.AA.AA.AA
nickname=Jane
```

5.3.15 DeleteWWNForLun

- Command example:

```
HiCommandCLI DeleteWwnForLun -o "D:/logs/9200 AddWwnForLun.log" "serialnum=0001"
"model=HDS9200" "port=16" "devnum=16" "domain=0" "wwn=11.55.77.99.33.00.22.66"
```

- Command result:

```
RESPONSE:
(Command completed; no data returned)
```

5.4 Host Management Commands

5.4.1 AddHost

- Command example:

```
HiCommandCLI AddHost -o "D:\logs\AddHost.log" "hostname=toro2" "  
wwnlist=AA.CC.CC.CC.CC.CC.CC.CC,00.CC.CC.CC.CC.CC.CC" "ipaddress=192.168.32.63"
```

- Command result:

```
RESPONSE:  
An instance of Host  
  objectID=HOST.279  
  name=hit  
  ipAddress=192.168.32.63  
  capacityInKB=0
```

5.4.2 DeleteHost

- Command example:

```
HiCommandCLI DeleteHost -o "D:\logs\ DeleteHost.log" "objectID=HOST.1"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

5.4.3 ModifyHost

- Command example:

```
HiCommandCLI ModifyHost -o "D:\logs\ModifyHost.log" "objectid=HOST.3" "hostname=snow"  
"wwnlist=12.34.56.78.90.AB.CD.EF,01.23.45.67.89.AB.CD.EF" "ipaddress=172.18.32.9"
```

- Command result:

```
RESPONSE:  
An instance of Host  
  objectID=HOST.3  
  name=snow  
  ipAddress=172.18.32.9  
  capacityInKB=0
```

5.4.4 GetHost

- Command example 1:

```
HiCommandCLI GetHost -o "D:\logs\GetHost.log" "objectID=HOST.1"
```

- Command result:

```
RESPONSE:
An instance of Host
  objectID=HOST.1
  name=toro2
  ipAddress=192.168.32.63
  capacityInKB=0
  List of 2 wwn elements:
    An instance of wwn
      wwn=00.99.66.33.88.55.22.77
      nickname=Jane
    An instance of wwn
      wwn=11.44.77.22.55.88.33.66
      nickname=Jone
```

- Command example 2:

```
HiCommandCLI GetHost -o "D:\logs\GetHost.log"
```

- Command result:

```
RESPONSE:
An instance of Host
  objectID=HOST.1
  name=toro2
  ipAddress=192.168.32.63
  capacityInKB=0
  List of 2 wwn elements:
    An instance of wwn
      wwn=00.99.66.33.88.55.22.77
      nickname=Jane
    An instance of wwn
      wwn=11.44.77.22.55.88.33.66
      nickname=Jone
.
. (repeated for other Host instances)
.
```

5.4.5 AddHostInfo

- Command example:

```
HiCommandCLI AddHostInfo -o "D:\logs\9980V AddHostInfo.log" "serialnum=0001"
"model=HDS9980V" "hostname=MY_HOSTINFO_HDS9980V" "ipaddress=172.16.64.159"
"mountpoint=C:" "port=7" "domain=0" "devnum=2" "osscsibus=0" "osscsiid=15" "oslun=1"
"portwnn=11.22.33.44.55.66.77.88" "filetype=NTFS" "filename=C:" "size=10"
"percentused=10"
```

- Command result:

```
RESPONSE:
An instance of HostInfo
  objectID=HOSTINFO.MY_HOSTINFO_HDS9980V.0.15.1
  name=MY_HOSTINFO_HDS9980V
  serialNumber=0001
  arrayType=HDS9980V
  ipAddress=172.16.64.159
  mountPoint=C:
  portID=7
  domainID=0
  scsiID=15
  lun=2
  devNum=2
  osScsiBus=0
  osScsiID=15
  osLun=1
  portWwn=11.22.33.44.55.66.77.88
  fileType=NTFS
  fileName=C:
  sizeInMB=10
  percentUsed=10
  lastUpdated=1039003476
```

5.4.6 DeleteHostInfo

- Command example:

```
HiCommandCLI DeleteHostInfo -o "D:\logs\9980V DeleteHostInfo.log"
"objectid=HOSTINFO.MY_HOSTINFO_HDS9980V.0.15.1"
```

- Command result:

```
RESPONSE:
(Command completed; no data returned)
```

5.4.7 ModifyHostInfo

- Command example:

```
HiCommandCLI ModifyHostInfo -o "D:\logs\9980V ModifyHostInfo.log"  
"objectId=HOSTINFO.MY_HOSTINFO_HDS9980V.0.15.1" "model=HDS9980V" "serialnum=0001"  
"hostname=MY_HOSTINFO_SANRISE9900V" "ipAddress=111.111.111.111" "mountPoint=E:"  
"port=7" "domain=0" "devnum=2" "portwwn=11.33.55.77.99.BB.DD.FF" "filetype=NTFS1"  
"filename=oo" "size=156" "percentused=50"
```

- Command result:

```
RESPONSE:  
An instance of HostInfo  
  objectID=HOSTINFO.MY_HOSTINFO_HDS9980V.0.15.1  
  name=MY_HOSTINFO_HDS9980V  
  serialNumber=0001  
  arrayType=HDS9980V  
  ipAddress=111.111.111.111  
  mountPoint=E:  
  portID=7  
  domainID=0  
  scsiID=15  
  lun=2  
  devNum=2  
  osScsiBus=0  
  osScsiID=15  
  osLun=1  
  portWWN=11.33.55.77.99.BB.DD.FF  
  fileType=NTFS1  
  fileName=oo  
  sizeInMB=156  
  percentUsed=50  
  lastUpdated=1039003476
```

5.4.8 GetHostInfo

- Command example 1:

```
HiCommandCLI GetHostInfo -o "D:\ logs\9980V GetHostInfo.log"  
"objectId=HOSTINFO.MY_HOSTINFO_HDS9980V.0.15.1"
```

- Command result:

```
RESPONSE:  
An instance of HostInfo  
  objectID=HOSTINFO.HIS_HOSTINFO_HDS9980V.0.15.1  
  name=HIS_HOSTINFO_HDS9980V  
  serialNumber=0001  
  arrayType=HDS9980V  
  ipAddress=172.16.64.159  
  mountPoint=C:  
  portID=7  
  domainID=0  
  scsiID=15  
  lun=3  
  devNum=3  
  osScsiBus=0  
  osScsiID=15  
  osLun=1  
  portWWN=11.22.33.44.55.66.77.88  
  fileSystemType=NTFS  
  fileSystemName=C:  
  sizeInMB=10  
  percentUsed=10  
  lastUpdated=1039003852
```

- Command example 2:

```
HiCommandCLI GetHostInfo -o "D:\ logs\9980V GetHostInfo.log"
```

- Command result:

```
RESPONSE:  
An instance of HostInfo  
  objectID=HOSTINFO.HIS_HOSTINFO_HDS9980V.0.15.1  
  name=HIS_HOSTINFO_HDS9980V  
  serialNumber=0001  
  arrayType=HDS9980V  
  ipAddress=172.16.64.159  
  mountPoint=C:  
  portID=7  
  domainID=0  
  scsiID=15  
  lun=3  
  devNum=3  
  osScsiBus=0  
  osScsiID=15  
  osLun=1  
  portWWN=11.22.33.44.55.66.77.88  
  fileSystemType=NTFS  
  fileSystemName=C:  
  sizeInMB=10  
  percentUsed=10  
  lastUpdated=1039003852  
... (repeated for other HostInfo instances)
```

5.5 Server Management Commands

5.5.1 GetServerInfo

- Command example 1:

```
HiCommandCLI GetServerInfo -o "D:\logs\GetServerInfo.log"
```

- Command result:

```
RESPONSE:
An instance of ServerInfo
  serverVersion=HiCommand Device Manager Server v2.2 Build 0220-05 (Oct 12, 2002)
  serverURL=http://localhost:2001
  upTime=3 hours 35 minutes 6 seconds
  upSince=Wed, 4 Dec 2002 08:13:49 GMT
  List of 7 StorageArray elements:
    An instance of StorageArray
      arrayFamily=HDS9900V
      displayArrayFamily=SANRISE H1024/H128
    An instance of StorageArray
      arrayFamily=HDS9900
      displayArrayFamily=SANRISE H512/H48
    An instance of StorageArray
      arrayFamily=HDS9200
      displayArrayFamily=SANRISE1000
    An instance of StorageArray
      arrayFamily=HDS9900
      displayArrayFamily=SANRISE2000
    An instance of StorageArray
      arrayFamily=HDS9500V
      displayArrayFamily=SANRISE9500V
    An instance of StorageArray
      arrayFamily=HDS9900V
      displayArrayFamily=SANRISE9900V
    An instance of StorageArray
      arrayFamily=SUNT3
      displayArrayFamily=SUNT3
```

5.5.2 GetLogFile

- Command example:

```
HiCommandCLI GetLogFile -o "D:\logs\GetLogFile.log" "filename=error.log"
```

- Command result:

```
RESPONSE:
An instance of File
  name=error.log
  ----- Contents of File Follow -----
    .
    . (Contents of File "error.log" is omitted here)
    .
  ----- End of File Contents - -----
```

5.5.3 ModifyDebugLevel

- Command example:

```
HiCommandCLI ModifyDebugLevel -o "D:\logs\ModifyDebugLevel.log" "level=0"
```

- Command result:

```
RESPONSE:  
An instance of DebugLevel  
  value=0 [Full Debugging trace]  
  description=Debugging Trace
```

5.5.4 GetDebugLevel

- Command example:

```
HiCommandCLI GetDebugLevel -o "D:\logs\GetDebugLevel.log"
```

- Command result:

```
RESPONSE:  
An instance of DebugLevel  
  value=0 [Full Debugging trace]  
  description=Debugging Trace
```

5.5.5 GetAlerts

- Command example:

```
HiCommandCLI GetAlerts -o "D:\logs\GetAlerts.log" "countfilter=2"  
"timefilter=2002/04/01 00:00:00"
```

- Command result:

```
RESPONSE:  
An instance of Alerts  
  .  
  . (Attributes of Alerts are omitted here)  
  .
```

5.5.6 DeleteAlerts

- Command example 1:

```
HiCommandCLI DeleteAlerts -o "D:\logs\ALL DeleteAlerts.log" "alertnum=6"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

- Command example 2:

```
HiCommandCLI DeleteAlerts -o "D:\logs\ALL DeleteAlerts.log"  
"source=ARRAY.HDS9200.0207"
```

- Command result:

```
RESPONSE:  
(Command completed; no data returned)
```

5.5.7 AddURLLink

- Command example:

```
HiCommandCLI AddURLLink -o "D:\logs\9980V AddURLLink.log" "url=192.168.99.AA"  
"name=AUTO" "linkedid=HSDOMAIN.HDS9980V.0001.0.3" "description=ARRAY.HDS9980V.0001"
```

- Command result:

```
RESPONSE:  
An instance of URLLink  
  objectID=URLLINK.HSDOMAIN.HDS9980V.0001.0.3 .1  
  name=AUTO  
  url=192.168.99.AA  
  linkedID=HSDOMAIN.HDS9980V.0001.0.3  
  description=ARRAY.HDS9980V.0001
```

5.5.8 DeleteURLLink

- Command example 1:

```
HiCommandCLI DeleteURLLink -o "D:\logs\9980V DeleteURLLink.log" "linkedid=HSDOMAIN.HDS9980V.0001.0.3"
```

- Command result:

```
RESPONSE:  
(Command completed; empty list returned)
```

- Command example 2:

```
HiCommandCLI DeleteURLLink -o "D:\logs\9980V DeleteURLLink.log" "objectid=URLLINK.HSDOMAIN.HDS9980V.0001.0.3.1"
```

- Command result:

```
RESPONSE:  
(Command completed; empty list returned)
```

5.5.9 GetURLLink

- Command example 1:

```
HiCommandCLI GetURLLink -o "D:\logs\9980V GetURLLink.log" "objectid=URLLINK.HSDOMAIN.HDS9980V.0001.0.3.1"
```

- Command result:

```
RESPONSE:  
An instance of URLLink  
  objectID=URLLINK.HSDOMAIN.HDS9980V.0001.0.3.1  
  name=AUTO  
  url=192.168.99.AA  
  linkedID=HSDOMAIN.HDS9980V.0001.0.3  
  description=ARRAY.HDS9980V.0001
```

- Command example 2:

```
HiCommandCLI GetURLLink -o "D:\logs\9980V GetURLLink.log" "linkedid=HSDOMAIN.HDS9980V.0001.0.3"
```

- Command result:

```
RESPONSE:  
An instance of URLLink  
  objectID=URLLINK.HSDOMAIN.HDS9980V.0001.0.3.1  
  name=AUTO  
  url=192.168.99.AA  
  linkedID=HSDOMAIN.HDS9980V.0001.0.3  
  description=ARRAY.HDS9980V.0001
```


Chapter 6 Troubleshooting

6.1 Troubleshooting

If there is a problem with the HiCommand™ Device Manager Command Line Interface:

- Make sure that the problem is not being caused by the PC or LAN hardware or software, and try restarting the PC.
- Make sure that the problem is not occurring at the HiCommand™ Device Manager Server. Please refer to the *HiCommand™ Device Manager Server Installation and Configuration Guide* (MK-91HC002) for troubleshooting information for the Device Manager Server.
- For troubleshooting information specific to the Device Manager CLI, see Table 6.1.
- For troubleshooting information common to the Device Manager CLI and Web Client, please refer to the *HiCommand™ Device Manager Web Client User's Guide*.
- If you need to call the Hitachi Data Systems Support Center, please see section 6.2 for instructions.

Important: Make sure to read the HiCommand™ Device Manager **ReadMe** file. Always make sure that only one HiCommand™ Device Manager Server at a time is actively managing a single subsystem.

Table 6.1 General Troubleshooting Information

Problem	Description and Recommended Action
CLI application does not run. Error message says: C:\HiCommand>HiCommandCLI help GetStorageArray The name specified is not recognized as an internal or external command, operable program or batch file	CAUSE: The most likely problem is that there is no java.exe file in your path. SOLUTION: Either update your path to include the directory holding the java.exe supplied by the Java™ JDK™ or JRE™ software, or edit the HiCommandCLI.bat file to specify the full path to your java.exe file.
HiCommand™ Device Manager CLI.bat script does not run. Error message says: C:\HiCommand>HiCommandCLI help GetStorageArray Exception in thread "main" Java.lang.NoClassDefFoundError: com.hitachi/hds/cli/Hicmd	CAUSE: The HiCommandCLI.jar file may have been moved or renamed. SOLUTION: Make sure that the HiCommandCLI.jar file has not been moved or renamed, and that you are executing from the directory where the HiCommandCLI.bat (or HiCommandCLI.sh, on UNIX® systems) and HiCommandCLI.jar files reside.

6.2 Hitachi Data Systems Support Center

If you need to call the Hitachi Data Systems Support Center, make sure to provide as much information about the problem as possible, including the circumstances surrounding the error or failure and the exact content of any error messages displayed on the HiCommand™ Device Manager system and/or host system.

The worldwide Hitachi Data Systems Support Centers are:

- Hitachi Data Systems North America/Latin America
San Diego, California, USA
1-800-348-4357
- Hitachi Data Systems Europe
Contact Hitachi Data Systems Local Support
- Hitachi Data Systems Asia Pacific
North Ryde, Australia
011-61-2-9325-3300

Acronyms and Abbreviations

API	application program interface
CLI	command-line interface
GB	gigabytes (1024 MB)
GUI	graphical user interface
HTML	hypertext markup language
HTTP	hypertext transmission protocol
HTTPS	hypertext transmission protocol secure
ID	identification, identifier
IP	internet protocol
JRE	Java™ Runtime Environment
KB	kilobytes (1024 bytes)
LDEV	logical device
LU	logical unit
LUN	logical unit number
LUSE	LUN Expansion, LU Size Expansion (feature of 9900V and 9900)
MB	megabytes (1024 KB)
OS	operating system
PDEV	physical device
RAID	redundant array of independent disks
SCSI	small computer system interface
SNMP	simple network management protocol
SRM	StorEdge Resource Management Suite software
SVP	service processor (component of 9900V and 9900)
URL	uniform resource locator
WWN	worldwide name
XML	extensible markup language

