



# StorageTek™ ACSL

**AUTOMATED CARTRIDGE SYSTEM LIBRARY SOFTWARE**

**INSTALLATION, CONFIGURATION, AND ADMINISTRATION**

312572307

Version: 7.1/7.1.1



# **Automated Cartridge System Library Software**

## **Installation, Configuration, and Administration Guide**

**Versions:**  
**7.1 for SPARC Solaris and AIX**  
**7.1.1 for x86 Solaris**

312572307



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

Sun Microsystems, Inc. has intellectual property rights relating to technology that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

This document and the product to which it pertains are distributed under licenses restricting their use, copying, distribution, and decompilation. No part of the product or of this document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

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

Sun, Sun Microsystems, the Sun logo, Java, AnswerBook2, docs.sun.com, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and in other countries.

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

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

StorageTek is a trademark of Storage Technology Corporation.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

U.S. Government Rights—Commercial use. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

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

Copyright 2006 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, Californie 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. a les droits de propriété intellectuels relatants à la technologie qui est décrit dans ce document. En particulier, et sans la limitation, ces droits de propriété intellectuels peuvent inclure un ou plus des brevets américains énumérés à <http://www.sun.com/patents> et un ou les brevets plus supplémentaires ou les applications de brevet en attente dans les Etats-Unis et dans les autres pays.

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

Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

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

Sun, Sun Microsystems, le logo Sun, Java, AnswerBook2, docs.sun.com, et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

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

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

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

We welcome your feedback. Please contact the Global Learning Solutions Feedback System at:

[GLSFS@Stortek.com](mailto:GLSFS@Stortek.com)

or

Sun Learning Services  
Sun Microsystems Inc.  
One StorageTek Drive  
Louisville, CO 80028-3256  
USA





# Summary of Changes

---

EC released document table

EC	Date	Edition	Revision	Description
128921	May 2004	First	A	Initial release.
132082	October 2004	Second	B	See this edition for details.
132225	February 2005	Third	C	See this edition for details.
132322	February 2006	Fourth	D	Added support for PUT0601.
132564	June 2006	Fifth	E	Added support (ACSL 7.1.1) for Solaris 10 x86.  This release <b>does not</b> support AIX and Solaris SPARC platforms. For these platforms, continue to use Solaris 7.1 with PUT0601S (Solaris) or PUT0601A (AIX).
132803	November 2006	Sixth	F	This release supports PUT0602S (7.1 Solaris SPARC), PUT0602A (7.1 AIX), and PUT0602X (7.1.1 Solaris 10 x86).
132982	March 2007	Seventh	G	This release supports PUT0701 for ACSLS 7.1/7.1.1.

## Summary of Changes

# Contents

---

<b>Summary of Changes</b> .....	<b>vii</b>
<b>Contents</b> .....	<b>ix</b>
<b>List of Figures</b> .....	<b>xxxiv</b>
<b>List of Tables</b> .....	<b>xxxvi</b>
<b>Preface</b> .....	<b>xxxvii</b>
About this Book .....	xxxvii
Audience .....	xxxvii
About the Software .....	xxxvii
Conventions for Reader Usability .....	xxxviii
Typographic .....	xxxviii
Keys .....	xxxix
Enter Command .....	xxxix
Symbols .....	xxxix
Technical Support .....	xxxix
Related Documentation .....	xxxix
ACSLs Documentation .....	xxxix
ACSLs Information on the StorageTek CRC .....	xl
ACS Tape Hardware Info on the StorageTek CRC .....	xl
<b>Customer Contacts</b> .....	<b>1</b>
Customer Support .....	1
Customer-initiated Maintenance .....	1
Sun's Worldwide Offices .....	2
<b>1: Overview</b> .....	<b>1</b>
What is ACSLS? .....	1
The acssa and acsss User IDs .....	2
Starting ACSLS .....	3
Stopping ACSLS .....	3
Using a cmd_proc .....	4
cmd_proc Window .....	4
How to Suspend and Resume a cmd_proc .....	6
How to Exit a cmd_proc .....	6
Manually Killing the ACSLS Process .....	7

Starting a cmd_proc . . . . .	7
Logging in as acsss from the ACSLS Server . . . . .	7
Logging in Remotely as acssa . . . . .	8
Logging in as acssa from a Non-UNIX Client . . . . .	8
Manually Starting a cmd_proc . . . . .	9
cmd_proc options . . . . .	9
cmd_proc Keyboard Shortcuts . . . . .	10
Redirecting cmd_proc Inputs and Outputs . . . . .	11
Idling ACSLS . . . . .	12
Restarting ACSLS . . . . .	12
ACSLS Directory Structure . . . . .	13
<b>2: ACSLS License Key. . . . .</b>	<b>21</b>
ACSLS License Key Utility . . . . .	21
Obtaining a Permanent Key . . . . .	22
Obtaining an Emergency Key . . . . .	22
License Key Format . . . . .	23
License Key Validation . . . . .	24
License Key Operational Procedures . . . . .	24
<b>3: Installing ACSLS 7.1 on Solaris 9. . . . .</b>	<b>27</b>
Step 1: Export the Database . . . . .	28
Step 2: Preparing for Solaris Installation . . . . .	28
Step 3: Installing the Solaris 9 Operating System . . . . .	29
Step 4: Preparing for ACSLS Installation . . . . .	36
Step 5: Installing ACSLS . . . . .	38
Step 6: Import the Database . . . . .	44
Step 7: Configure your Library Hardware . . . . .	45
Step 8: Enter your License Key Information . . . . .	45
Step 9: Reset any Custom Dynamic or Static Variables . . . . .	45
Step 10: Configure the Second Disk . . . . .	45
Step 11: Audit new libraries . . . . .	45
Step 12: Verify ACSLS Installation . . . . .	46
ACSLS Setup for Client Communication . . . . .	47
Regressing to a Previous Version of ACSLS . . . . .	47
Uninstalling ACSLS 7.1 . . . . .	47
<b>4: Installing ACSLS 7.1 on Solaris 10. . . . .</b>	<b>49</b>
Step 1: Export the Database . . . . .	50
Step 2: Prepare for Solaris Installation . . . . .	50
Pre-Installation Checklist . . . . .	50
Step 3: Install the Solaris 10 Operating System . . . . .	54
Step 4: Prepare for ACSLS Installation . . . . .	57

Step 5: Install ACSLS	59
Remove any previous version	59
Install pkgadd	62
Install ACSLS software	63
Step 6: Import the Database	66
Step 7: Configure your Library Hardware	66
Step 8: Enter your License Key Information	66
Step 9: Reset any Custom Dynamic or Static Variables	66
Step 10: Configure the Second Disk	67
Step 11: Audit new libraries	67
Step 12: Verify ACSLS Installation	67
ACSLS Setup for Client Communication	68
Regressing to a Previous Version of ACSLS	68
Uninstalling ACSLS 7.1	69
Running ACSLS within Solaris Zones	70
Installing the STKchanger	71
Connecting to a library	72
<b>5: Installing ACSLS 7.1.1 on Solaris 10 x86.</b>	<b>75</b>
Step 1: Export the Database	76
Step 2: Preparing for Solaris Installation	76
Pre-Installation Checklist	76
Step 3: Install Solaris 10 Operation System for x86 Platforms	81
Step 4: Prepare for ACSLS Installation	89
Step 5: Install ACSLS	91
Initiate the installation shell script	92
Step 6: Option: Import the Database	94
Step 7: Configure your Library Hardware	94
Step 8: Enter your License Key Information	94
Step 9: Reset any Custom Dynamic or Static Variables	95
Step 10: Configure the Second Disk	95
Step 11: Audit new libraries	95
Step 12: Verify ACSLS Installation	95
ACSLS Setup for Client Communication	96
Uninstalling ACSLS 7.1.1	96
<b>6: Installing ACSLS 7.1 on AIX</b>	<b>99</b>
Step 1: Export the Database	100
Step 2: Prepare for AIX Installation	100
Step 3: Prepare for ACSLS Installation	101
Step 4: Install ACSLS	103
Step 5: Import the Database	109

Step 6: Configure your Library Hardware .....	109
Step 7: Enter your License Key Information .....	110
Step 8: Reset any Custom Dynamic or Static Variables .....	110
Step 9: Configure the Second Disk .....	110
Step 10: Audit New Libraries .....	110
Step 11: Verify ACSLS Installation .....	110
ACSLs Setup for Client Communication .....	111
Regressing to a Previous Version of ACSLS .....	112
Uninstalling ACSLS 7.1 .....	112
<b>7: ACSLS Configuration .....</b>	<b>115</b>
Overview .....	115
Accessing the ACSLS Feature Configuration Menu .....	116
Verifying and Changing Dynamic and Static Variables .....	117
Setting CSI Tuning Variables .....	117
Setting Event Logging Variables .....	120
Setting General Product Behavior Variables .....	121
Setting Access Control Variables .....	127
Setting Automatic Backup Variables .....	128
Rebuilding Access Control Information .....	129
Defining Event Notification Settings .....	129
Configuring or Updating your Library Hardware .....	130
<b>8: Access Control .....</b>	<b>135</b>
How ACSLS Access Control Works .....	135
Setting up Access Control Variables .....	136
Creating a client-identification file for the corresponding network protocol .....	137
Creating an internet.addresses File for TCP/IP Protocol Clients .....	137
Creating an adi.names File for OSLAN Protocol Networks .....	138
Creating an lu62.names File for SNA Protocol Clients .....	139
Command Access File Types .....	140
Creating Command Allow and/or Disallow Files .....	140
Command Access Control Reference .....	142
Volume Access Control .....	143
Creating Users Allow and Disallow Files .....	143
Using Allow and Disallow Templates .....	143
Volume Access Control Commands .....	146
Setting Volume Ownership .....	147
Access Control Logging .....	151
Enabling ACSLS Access Control .....	151
Enabling ACSLS access control: .....	151
Access Control Scenarios .....	152

Command Access Control Scenarios .....	152
Command Access Control Scenario 1 .....	153
Command Access Control Scenario 2 .....	153
Command Access Control Scenario 3 .....	153
Volume Access Control Scenarios .....	155
Volume Access Control Scenario 1 .....	155
Volume Access Control Scenario 2 .....	155
Volume Access Control Scenario 3 .....	155
Volume Access Control Scenario 4 .....	156
Volume Access Control Scenario 5 .....	157
Maintaining and Modifying ACSLS Access Control .....	158
Logging Access Control Messages .....	158
Reporting on Volume Ownership .....	158
Error Conditions .....	160
Modifying or Adding Access Control Files .....	161
Rebuilding Access Control Information .....	161
<b>9: Library Management .....</b>	<b>163</b>
Auditing the Library .....	163
When to Run an Audit .....	163
Audit Intervals .....	164
Auditing the SL8500 Library .....	165
Auditing the SL500 Library .....	165
Adding a SCSI or fibre-attached library to ACSLS .....	165
Creating an mchanger Device Node .....	166
Removing an mchanger Device Node .....	166
Using the Extended Store Feature .....	167
Managing a Mixed-Media Library .....	169
Displaying the ACSLS Mixed-Media Settings .....	170
Using Mixed-Media Support with 3490E Cartridges .....	170
Mixed-Media Restrictions for SCSI-Attached LSMs .....	171
Setting Scratch Preferences .....	172
User-Defined Mixed-Media Files .....	172
System-Defined Mixed-Media Files .....	173
How ACSLS Uses the Mixed-Media Files .....	173
Defining a Scratch Preferences File .....	174
Managing a Dual-LMU Configuration .....	175
Managing a Dual-LAN Client Configuration .....	177
Registering the IP Addresses of Primary and Secondary LANs .....	178
Installing a Second Ethernet Port for a Multi-Homed ACSLS Server .....	178
Setting the TCP/IP Connection Timeout Interval .....	179
Registering for Event Notification .....	180

<b>10: Cartridge Management</b> .....	<b>181</b>
Using the Manual Volume Delete Utility .....	181
Using Absent and Ejected Cartridge Support .....	182
Absent, Ejected and Missing Cartridges .....	183
Cartridge (Volume) Status Reporting .....	183
Cartridge Recovery .....	184
Missing Cartridges .....	185
Absent and Ejected Cartridges .....	186
Cartridges Not Found .....	186
Cartridges Found .....	186
Populating the LSM .....	186
Using the CAP .....	187
CAP Types .....	187
CAP States .....	188
CAP Modes .....	189
CAP Priorities .....	191
Displaying CAP Information .....	191
CAPs Must be Empty Before Enters and Ejects Terminate .....	192
Entering Cartridges .....	192
Terminating an Enter Request .....	194
Ejecting Cartridges .....	194
Managing Scratch Cartridges .....	195
Displaying Scratch Pool and Scratch Cartridge Information .....	195
Adding Scratch Cartridges to the Library .....	196
Rebalancing Scratch Pools .....	196
Deleting Scratch Pools .....	197
Emptying a Scratch Pool .....	197
Deleting a Single Pool .....	197
Deleting All Empty Pools .....	197
Mounting Scratch Cartridges .....	197
Single-Media Environments .....	198
Mixed-Media Environments .....	198
Unscratching Cartridges .....	198
Cleaning Cartridges .....	199
Automatic Cleaning .....	199
Defining Cleaning Cartridges .....	200
Ejecting Used Cleaning Cartridges .....	201
Manually Cleaning a Transport .....	201
Cleaning Cartridges on SCSI-Attached LSMs .....	202
Correcting Cleaning Cartridge Problems .....	202
Manually Loading Cartridges Into Drives In a Disabled LSM .....	203

<b>11: Database Backup and Restore</b> .....	<b>205</b>
Utilities Used .....	206
Exporting the Database .....	206
Exporting to a Disk File .....	207
Exporting to Tape .....	208
Importing the Database .....	209
Importing From a Disk File .....	210
Importing from Tape .....	213
Verifying the Imported Database and Library Configuration .....	214
Automatic Database Backup .....	215
Performing Manual Backups to Tape .....	216
Backing up to the Default Tape Device Attached to the ACSLS Server .....	216
Backing up to a Specified Tape Device Attached to the ACSLS Server .....	217
Backing up to a UNIX File .....	218
Backing up ACSLS and Restoring to Another Server .....	218
Backup Process .....	218
Restore the Backup to a Different Server .....	219
Recovering and Restoring .....	220
Restoring the Database to the Most Recent Backup .....	222
Restoring the Database to a Specified Date and Time .....	223
Recovering from a Primary Disk Failure .....	225
Recovering from a Second Disk Failure .....	226
Recovering from a Failed Server .....	227
Restoring Miscellaneous ACSLS Files .....	229
Restarting the Database .....	231
<b>12: Reporting and Logging</b> .....	<b>233</b>
Creating a Custom Volume Report .....	233
Custom Volume Report .....	234
Custom Volume Report Example 2 .....	235
Creating a Logging Volume Statistics Report .....	236
<b>13: Utility Reference</b> .....	<b>241</b>
Using the ACSLS Utilities .....	243
bdb.acsss .....	243
Format .....	243
Options .....	243
Usage .....	244
cmd_proc_shell .....	245
Format .....	245
Dynamic Configuration (config) utilities .....	245
Important .....	246

Limitations of Dynamic Configuration . . . . .	246
Solution: . . . . .	247
Things You Should Not Do . . . . .	247
config acs . . . . .	247
Format . . . . .	247
Adding a new ACS . . . . .	247
Reconfiguring an existing ACS . . . . .	248
Limitation of config acs . . . . .	249
config drives . . . . .	249
Format . . . . .	249
Usage . . . . .	249
config lsm . . . . .	250
Format . . . . .	250
Usage . . . . .	250
Limitation of config lsm . . . . .	251
config ports . . . . .	251
Format . . . . .	251
Usage . . . . .	251
Limitations of config ports . . . . .	252
db_command . . . . .	252
Format . . . . .	252
Options . . . . .	252
Usage . . . . .	253
db_export.sh . . . . .	253
Format . . . . .	253
Options . . . . .	253
Usage . . . . .	253
Examples . . . . .	253
db_import.sh . . . . .	254
Format . . . . .	254
Options . . . . .	254
Usage . . . . .	254
del_vol . . . . .	254
Format . . . . .	255
Options . . . . .	255
Usage . . . . .	255
Example . . . . .	256
See Also . . . . .	256
drives_media.sh . . . . .	256
Format . . . . .	256
Options . . . . .	256
Example . . . . .	256

ejecting.sh	260
Format	260
Options	260
Usage	260
Examples	260
find_tty.sh	261
Format	261
Options	261
Usage	261
Example	262
free_cells.sh	262
Format	262
Option	262
Examples	263
Free cells in each LSM	263
Free cells in each ACS	263
Free cells managed by the ACSLS server	264
..... get_license_info	265
Format	265
Options	265
Example	266
greplog	266
Format	266
Options	266
Usage	266
kill.acsss	267
Format	267
Options	267
Usage	267
See also	267
moving.sh	267
Format	268
Usage	268
Creating the Volume List File	269
Procedures for Moving a Group of Cartridges	269
Moving Cartridges Before Removing an LSM from an ACS	269
Moving Cartridges Before Changing or Removing Panels	271
Examples	272
Managing Cartridges for Performance	273
See also	274
probeFibre.sh	274
Format	274

rc.acsss	274
Format	274
Options	274
Usage	274
Examples	275
See Also	275
rdb.acsss	275
Format	275
Menu Options	276
See Also	277
showDevs.sh	278
Format	278
Usage	278
Options	278
Examples	279
stats_report	280
Format	280
Usage	281
volrpt	281
Format	281
Options	281
Usage	282
Examples	283
Notes	284
See Also	286
watch_vols	286
Format	286
Usage	286
Examples	287
<b>14: Command References</b>	<b>289</b>
General Command Syntax	290
Component Types and Identifiers	291
Common Command Error Messages	294
audit	294
Format	295
Usage	296
Examples	297
Notes	297
See Also	298
Command Area Messages	298
Display Area Messages	300

cancel	300
Format	300
Options	300
Examples	302
Notes	303
See Also	303
Command Area Messages	303
<b>Display Area Messages</b>	<b>304</b>
clear lock	304
Format	304
Options	304
Usage	304
Examples	304
Notes	305
See Also	305
Command Area Messages	305
Display Area Messages	306
define pool	306
Format	306
Options	306
Usage	306
Examples	306
Notes	307
See Also	307
Command Area Messages	307
Display Area Messages	307
delete pool	308
Format	308
Options	308
Usage	308
Examples	308
Notes	309
See Also	309
Command Area Messages	309
dismount	310
Format	310
Options	310
Usage	310
Examples	311
Notes	311
See Also	312
Command Area Messages	312

	Display Area Messages .....	314
eject .....		314
	Format .....	314
	Options .....	314
	Usage .....	314
	Examples .....	315
	Notes .....	315
	See Also .....	316
	Command Area Messages .....	316
	Display Area Messages .....	318
enter .....		318
	Options .....	319
	Usage .....	319
	Examples .....	319
	Notes .....	319
	Command Area Messages .....	320
	Display Area Messages .....	321
idle .....		322
	Format .....	322
	Options .....	322
	Usage .....	322
	Examples .....	323
	Notes .....	323
	See Also .....	323
	Command Area Messages .....	323
	Display Area Messages .....	323
lock .....		324
	Format .....	324
	Options .....	324
	Usage .....	324
	Examples .....	325
	Notes .....	325
	See Also .....	325
	Command Area Messages .....	325
	Display Area Messages .....	326
logoff .....		326
	Format .....	326
	Options .....	326
	Usage .....	326
	Examples .....	326
	Notes .....	327
	See Also .....	327

Command Area Messages . . . . .	327
Display Area Messages . . . . .	327
mount . . . . .	327
Format . . . . .	327
Options . . . . .	327
Usage . . . . .	328
Examples . . . . .	328
Notes . . . . .	328
See Also . . . . .	329
Command Area Messages . . . . .	329
Display Area Messages . . . . .	330
mount * . . . . .	330
Format . . . . .	330
Options . . . . .	331
Usage . . . . .	331
Examples . . . . .	332
Notes . . . . .	333
See Also . . . . .	333
Command Area Messages . . . . .	334
<b>Display Area Messages . . . . .</b>	<b>335</b>
move . . . . .	335
Format . . . . .	335
Options . . . . .	336
Usage . . . . .	336
Examples . . . . .	336
Notes . . . . .	336
See Also . . . . .	337
Command Area Messages . . . . .	337
Display Area Messages . . . . .	337
query commands . . . . .	337
Format . . . . .	337
Notes . . . . .	338
See Also . . . . .	338
Command Area Messages . . . . .	338
Display Area Messages . . . . .	338
query acs . . . . .	338
Format . . . . .	338
Options . . . . .	339
Usage . . . . .	339
Examples . . . . .	340
See Also . . . . .	340
query cap . . . . .	340

Format	340
Options	340
Usage	340
Examples	342
See Also	342
query clean	342
Format	342
Options	342
Usage	342
Examples	343
See Also	343
query drive	343
Format	343
Options	344
Usage	344
Examples	345
See Also	345
query lmu	345
Format	345
Options	345
Usage	346
Examples	347
See Also	347
query lock	347
Format	347
Options	347
Usage	348
Examples	348
See Also	349
query lsm	349
Format	349
Options	349
Examples	350
See Also	350
query mount	350
Format	351
Options	351
Usage	351
Examples	352
See Also	353
query mount *	353
Format	353

Options	353
Usage	353
Examples	355
See Also	355
query pool	355
Format	355
Options	355
Usage	356
Examples	356
See Also	357
query port	357
Format	357
Options	357
Usage	357
Examples	358
See Also	358
query request	358
Format	358
Options	358
Usage	358
Examples	359
See Also	359
query scratch	359
Format	359
Options	359
Usage	360
Examples	360
See Also	361
query server	361
Format	361
Options	361
Usage	361
Examples	362
See Also	362
query cartridge	362
Format	362
Options	362
Usage	363
Examples	363
See Also	364
set commands	364
Format	364

See Also . . . . .	364
Command Area Messages . . . . .	364
Display Area Messages . . . . .	365
set cap mode . . . . .	365
Format . . . . .	365
Options . . . . .	365
Usage . . . . .	365
Examples . . . . .	365
See Also . . . . .	366
Command Area Messages . . . . .	366
set cap priority . . . . .	367
Format . . . . .	367
Options . . . . .	367
Usage . . . . .	367
Examples . . . . .	367
See Also . . . . .	368
Command Area Messages . . . . .	368
Display Area Messages . . . . .	368
set clean . . . . .	369
Format . . . . .	369
Options . . . . .	369
Usage . . . . .	369
Examples . . . . .	369
See Also . . . . .	369
Command Area Messages . . . . .	369
Display Area Messages . . . . .	370
set lock . . . . .	370
Format . . . . .	370
Options . . . . .	370
Usage . . . . .	370
Examples . . . . .	371
See Also . . . . .	371
Command Area Messages . . . . .	371
Display Area Messages . . . . .	371
set owner . . . . .	372
Format . . . . .	372
Options . . . . .	372
Usage . . . . .	372
Examples . . . . .	372
Notes . . . . .	372
See Also . . . . .	372
Command Area Messages . . . . .	372

	Display Area Messages	372
set scratch		372
	Format	372
	Options	373
	Usage	373
	Examples	373
	See Also	373
	Command Area Messages	374
	Display Area Messages	374
show		375
	Format	375
	Options	375
	Usage	375
	Examples	375
	Notes	375
	See Also	375
	Command Area Messages	376
	Display Area Messages	376
start		376
	Format	376
	Options	376
	Usage	376
	Examples	376
	See Also	377
	Command Area Messages	377
	Display Area Messages	377
switch Imu		377
	Format	377
	Options	378
	Usage	378
	Examples	378
	See Also	379
	Command Area Messages	379
	Display Area Messages	380
unlock		380
	Format	380
	Options	380
	Usage	381
	Examples	381
	See Also	381
	Command Area Messages	381
	Display Area Messages	382

vary	382
Format	382
Options	382
Usage	383
Examples	385
Notes	385
See Also	385
Command Area Messages	385
Display Area Messages	388
venter	388
Format	388
Options	388
Usage	389
Examples	389
See Also	390
Command Area Messages	390
Display Area Messages	391
<b>15: Display Command References</b>	<b>393</b>
display Commands	394
Format	395
display cap	396
Format	396
Fields	396
Options	396
Examples	398
display cell	398
Format	398
Fields	398
Options	398
Example	399
display drive	399
Format	399
Fields	399
Options	400
Examples	401
display lock	401
Format	401
Fields	402
Options	402
Example	403
display lsm	403

Format	403
Fields	403
Example	404
display panel	405
Format	405
Fields	405
Options	405
Example	406
display pool	406
Format	406
Fields	406
Options	406
Example	408
display port	408
Format	408
Fields	408
Example	409
display volume	409
Format	409
Fields	410
Options	410
Examples	412
<b>A: Second Disk Support</b>	<b>413</b>
Configuring the Second Disk on Solaris	413
Space Required by ACSLS on the Second Disk	413
Partitioning the Second Disk	414
Creating the File Systems	418
Installing ACSLS Second Disk Support	419
Restarting ACSLS	420
De-installing a Second Disk on Solaris	421
Configuring the Second Disk on AIX	422
Partitioning the Second Disk	422
Creating Second Disk File Systems	423
Mounting Second Disk File Systems	424
Creating Second Disk Backup File Systems	424
Mounting Second Disk Backup File Systems	425
Installing Second Disk Support	426
Restarting ACSLS	427
Importing the Database	427
Verifying the Imported Database and Library Configuration	428
Verifying ACSLS Software	430

Verifying ACSLS . . . . .	430
Running ivp.sh to Verify ACSLS . . . . .	431
ACSLS Setup for Client Communication . . . . .	432
Regressing to a Previous Version of ACSLS . . . . .	432
De-Installing a Second Disk on AIX . . . . .	432
<b>B: ACSLS Backup and Recovery Tools . . . . .</b>	<b>435</b>
ACSLS Backup Tools . . . . .	435
Automatic Backups . . . . .	435
Manual Backups . . . . .	436
Manual Database Exports . . . . .	437
ACSLS Recovery Tools . . . . .	438
Using rdb.acsss . . . . .	438
Restoring from current local disk backup . . . . .	438
Restoring from previous local disk backup (to a point-in-time) . . . . .	438
Restoring from a previous tape or network file backup . . . . .	439
Restoring a backup created on a different server . . . . .	439
Restoring ACSLS non-database files . . . . .	439
Using db_import.sh . . . . .	439
Importing database only from either a different platform or from an earlier release 440	
Importing miscellaneous configuration files exported from a different platform or from an earlier release . . . . .	440
Recovering a previous environment on the same platform and version (Disaster Recovery) . . . . .	440
Merging customized dynamic variables from a previous installation . . . . .	441
Disaster Scenarios . . . . .	441
Database becomes corrupted – need to restore the database to a working state . . . . .	441
Ran acsss_config against the wrong library . . . . .	441
Disk failure on primary disk with /export/home – second disk support installed . . . . .	442
Disk failure on /second_disk – second disk installed . . . . .	443
Server failure – rebuilding the same server with new hardware . . . . .	443
Server failure – rebuilding another ACSLS server with new hardware . . . . .	444
<b>C: ACSLS Support of the SL8500 . . . . .</b>	<b>445</b>
Before Configuring ACSLS for the SL8500 . . . . .	446
Connecting the SL8500 to ACSLS . . . . .	446
Verifying that all SL8500 Components are Operational . . . . .	447
Understanding SL8500 Internal Addresses and ACSLS Addresses . . . . .	447
Using SLConsole to Translate Addresses . . . . .	449
Tape Drive Locations . . . . .	451
SL8500 CAP Behavior . . . . .	453
Enter or Eject Operations . . . . .	453

Minimizing Elevator and PTP Activity . . . . .	454
Configuring Tape Drives to Support Your Workloads . . . . .	455
Managing Cartridge Locations . . . . .	456
Finding Missing Cartridges . . . . .	457
Varying the SL8500 Offline . . . . .	457
Use ACSLS to vary SL8500 components offline, not SLConsole . . . . .	457
When to vary SL8500 components offline to ACSLS . . . . .	458
Before opening the access door . . . . .	458
If an LSM (rail) is inoperative . . . . .	458
If a CAP is inoperative . . . . .	458
When closing the Service Safety Door . . . . .	458
When using the Service Safety Door do not use these ACSLS commands and utilities . . . . .	459
Using the Dynamic Configuration (config) utility . . . . .	460
Work around when an inoperative HandBot is adjacent to the middle SL8500 CAP magazine . . . . .	461
Using ACSLS HA . . . . .	462
Auditing the SL8500 After Expansion . . . . .	463
Auditing New Cartridge Addresses . . . . .	464
Procedure for Auditing an Expanded SL8500 . . . . .	465
Connecting SL8500s with Pass-Thru-Ports . . . . .	466
Before You Install SL8500 PTP Connections . . . . .	467
Adding New SL8500s . . . . .	467
Adding a New SL8500 to the Left . . . . .	468
Dynamically Configuring the New ACSLS Configuration . . . . .	468
Adding a New SL8500 to the Right . . . . .	469
Considerations When Adding an SL8500 to the Right . . . . .	469
Dynamically Configuring the New ACSLS Configuration . . . . .	469
Procedures for Merging ACSs . . . . .	471
Merging ACSs Numbered from Right to Left . . . . .	472
. . . . .	472
Procedure for Merging ACSs Numbered from Right to Left . . . . .	472
Merging ACSs Numbered from Left to Right . . . . .	473
Procedure for Merging ACSs Numbered from Left to Right . . . . .	473
Removing PTPs and Splitting ACSs . . . . .	474
Splitting ACSs, with the new ACS Created from SL8500(s) on the Left - Possible Scenarios . . . . .	475
ACSLs Procedure for Splitting an ACS . . . . .	475
Adding the New ACS on the Right Hand side of the Split . . . . .	476
Updating ACSLS After Adding HandBots . . . . .	477
ACSLs Dual TCP/IP Support on SL8500 . . . . .	478
Scenario 1 . . . . .	479

Scenario 2	481
Scenario 3	483
Scenario 4	484
Scenario 5	484
.....	485
Retaining Customized Routing Table Entries after a Reboot	486
Configuring ACSLS To Initialize Automatically At System Boot Time	487
Adding Custom Route To Be Initialized At Boot Time	487
Removing routing commands	488
<b>D: Library Partitioning</b>	<b>489</b>
Software Requirements	489
Guidelines for Partitioning	489
Plan Your New Configuration	490
Minimize Disruptions when Partitioning an SL8500	490
Changes in Configuration	490
Keeping Outages to a Minimum	490
Challenges	491
Partitioning a Library or Changing Partition IDs	491
Re-partitioning a Library	492
Changing a Partitioned ACS to an Un-partitioned ACS	493
CAP Behavior	494
Sharing CAPs in a Partitioned Library	494
When Another Host Needs the CAP	494
Reserving the CAP	495
Terminating a CAP Reservation	495
<b>E: ACSLS Support of the SL500</b>	<b>497</b>
ACSLs Connectivity	497
ACSLs and SL500 Library Differences	497
Library Configuration	497
Identifying Library Locations	497
Addressing schemes	497
Panels	498
Row numbers	498
Column numbers	498
Drive addresses	498
ACSLs Limitations	499
Configuring SL500 Library Settings	499
SL500 CAP Behavior	500
Auditing the Library	500
After adding, removing, or swapping modules	501

After manually adding or removing cartridges through the door .....	501
After entering or ejecting cartridges through the CAP .....	501
After changing library configuration settings .....	501
<b>F: Troubleshooting.....</b>	<b>503</b>
ACSLS and Library Hardware Error Recovery .....	503
Communications Failures .....	503
Hardware Failures .....	504
Software Failures .....	504
Tracking Software Problems .....	505
Applying Software Maintenance .....	505
ACSLS Event Log .....	505
Using the Event Log .....	505
Managing the Event Log .....	506
Using greplog to Search Even Logs .....	506
Format .....	506
Options .....	507
Usage .....	507
Examples .....	507
Note .....	507
Logs of ACSLS Installation and Utilities .....	508
Troubleshooting Library Connections .....	509
Part 1: ACSLS and OS Diagnostic Utilities .....	509
ck_port_node.sh .....	509
Example .....	509
probeFibre.sh .....	509
probescsi.sh .....	510
Example .....	510
showDevs.sh .....	510
testports .....	510
Example .....	511
testlmu .....	511
Example .....	511
System Utilities: AIX .....	511
lsdev .....	511
Example .....	511
Example (SCSI) .....	511
diag .....	512
Example .....	513
System Utilities: Solaris .....	513
probe-scsi-all .....	513
Example .....	514

Part 2: Strategies for Troubleshooting Host-Library Communication Problems . . .	514
Troubleshooting a Serial Connection . . . . .	514
Troubleshooting a SCSI Connection . . . . .	516
Gathering ACSLS Diagnostic Information . . . . .	518
<b>G: Improving Library Performance . . . . .</b>	<b>521</b>
Overview . . . . .	521
Minimizing Pass-thru Activity Between LSMs . . . . .	521
Floating Cartridges During a Dismount . . . . .	521
Entering Cartridges into a Library . . . . .	522
Ejecting Cartridges from a Library . . . . .	522
Maintaining Empty Storage Cells in an LSM . . . . .	522
Minimizing Pass-Thru Activity Using ACSAPI Requests and ACSLS Commands . . .	522
mount scratch and mount * . . . . .	523
query mount . . . . .	523
query mount scratch and query mount * . . . . .	523
Using these ACSAPI Requests & ACSLS Commands . . . . .	524
<b>H: Firewall Security Option . . . . .</b>	<b>527</b>
Overview . . . . .	527
Running ACSLS behind a Firewall . . . . .	527
Addressing security areas . . . . .	528
RPC . . . . .	528
Security . . . . .	528
Communications Components . . . . .	528
Benefits of the Firewall-Secure Option . . . . .	529
Server Side . . . . .	529
Server Port Restriction . . . . .	529
Client Side . . . . .	530
Client Port Restriction . . . . .	530
Advantages . . . . .	531
Firewall-Secure Solution Scenarios . . . . .	531
Example 1 . . . . .	532
Example 2 . . . . .	533
Example 3 . . . . .	534
Example 4 . . . . .	535
Turning On the Firewall-Secure Option . . . . .	535
Turning Off the Firewall-Secure Option . . . . .	538
Firewall-Secure Configuration . . . . .	539
Example . . . . .	539
Troubleshooting Firewall-Secure Communications . . . . .	540
Frequently Asked Questions . . . . .	542

<b>I: Media Management</b> .....	<b>547</b>
Overview .....	547
Problems .....	547
Examples .....	547
Solution .....	548
Managing Common Media Recorded in Different Densities with ACSLS Pools .....	549
<b>J: Installing AIX from a Backup</b> .....	<b>553</b>
Installing AIX from a System Backup .....	553
<b>K: CSCI</b> .....	<b>555</b>
Overview of CSCI .....	555
CSCI Connectivity .....	555
Architecture and Subsystems .....	555
Error Messages .....	557
Common Error Messages .....	557
Environmental Variables .....	558
<b>Glossary</b> .....	<b>561</b>
<b>Index</b> .....	<b>569</b>

# List of Figures

---

Figure 1. Library with ACSLS Server . . . . .	1
Figure 2. Example Screen Display with multiple windows . . . . .	2
Figure 3. cmd_proc Window . . . . .	5
Figure 4. Primary Disk Partitions Guidelines . . . . .	33
Figure 5. Primary Disk Partitions Guidelines . . . . .	34
Figure 6. Dual LMU Connection Configuration . . . . .	36
Figure 7. Primary Disk Partitions Guidelines . . . . .	56
Figure 8. Dual LMU Connection Configuration . . . . .	57
Figure 9. Primary Disk Partitions Guidelines . . . . .	88
Figure 10. AIX Requirements . . . . .	100
Figure 11. Client Setup Configuration Worksheet . . . . .	101
Figure 12. Example of internet.addresses file . . . . .	138
Figure 13. Example of adi.names file . . . . .	139
Figure 14. Example of u62.names file . . . . .	140
Figure 15. Sample command.ALL.allow file . . . . .	141
Figure 16. Sample command.START.allow file . . . . .	142
Figure 17. users.ALL.allow file . . . . .	145
Figure 18. users.EJECT.allow file . . . . .	146
Figure 19. Example of ownership.assignment file . . . . .	149
Figure 20. Sample internet.addresses file . . . . .	152
Figure 21. Sample command.START.allow file . . . . .	153
Figure 22. Sample command.EJECT.disallow file . . . . .	153
Figure 23. Sample command.ALL.allow file . . . . .	154
Figure 24. allow Files Names and Entries . . . . .	154
Figure 25. Sample users.MOUNT.allow file . . . . .	155
Figure 26. Sample users.ALL.allow file . . . . .	156
Figure 27. Sample users.MOUNT_READONLY.allow file . . . . .	156
Figure 28. Sample users.COMMAND.allow file . . . . .	157
Figure 29. Sample users.ALL.allow file . . . . .	157
Figure 30. Sample owner_id.volrpt File . . . . .	159
Figure 31. Sample owner_id.volrpt Using -f Option . . . . .	160
Figure 32. Sample Extended Store Feature Control File . . . . .	168
Figure 33. Modified Extended Store Feature Control File . . . . .	169
Figure 34. scratch_preferences.SAMPLE . . . . .	174
Figure 35. ACSLS Server in Dual-LMU Configuration . . . . .	176
Figure 36. Example of csc_ip_switch.dat File . . . . .	178
Figure 37. Example of Query CAP command . . . . .	192

Figure 38. Sample Volrpt Input File .....	234
Figure 39. Customized Volume Report Using Input File .....	235
Figure 40. Sample volrpt Control Statements .....	235
Figure 41. Customized volrpt Showing Absent and Ejected Volumes .....	236
Figure 42. Example of Volume Statistics Log File Entries .....	237
Figure 43. rdb.acsss Menu Options .....	276
Figure 44. Standard Volume Report .....	283
Figure 45. Format Menu .....	415
Figure 46. Partition Menu .....	415
Figure 47. Partition Table .....	416
Figure 48. Second Disk Partition Table .....	417
Figure 49. Partition Menu .....	418
Figure 50. SL8500 with an ACSLS Server .....	446
Figure 51. SL8500 Customer Expansion Module .....	464
Figure 52. Four connected SL8500 Libraries .....	466
Figure 53. Four connected SL8500 Libraries .....	468
Figure 54. Original Configuration Before an SL8500 is added to the Right .....	471
Figure 55. Correct Audit Order (shown in bold) and New Configuration .....	471
Figure 56. Existing Configuration – Three Separate ACSs .....	472
Figure 57. Desired Configuration – Single ACS .....	472
Figure 58. Existing Configuration - Three Separate ACSs .....	473
Figure 59. Desired Configuration - Single ACS .....	473
Figure 60. Existing Configuration – One ACS .....	475
Figure 61. Desired Configuration - Two ACSs .....	475
Figure 62. Example of drive addressing .....	499
Figure 63. Firewall security on ACSLS server side (CSI) only .....	532
Figure 64. Firewall Security on Client Server Side (SSI) Only .....	533
Figure 65. Firewall Security on Client (SSI) and ACSLS Server (CSI) using Portmapper ..	534
Figure 66. Firewall Security on Client (SSI) and ACSLS Server (CSI) Without the Portmapper Feature .....	535
Figure 67. CSCI Architecture and Subsystems .....	556
Figure 68. CSCI System with Token Ring Interface .....	556

# List of Tables

---

Table 1. cmd_proc Keyboard Shortcuts . . . . .	10
Table 2. ACSLS Directory Structure . . . . .	13
Table 3. Pre-Installation Worksheet . . . . .	28
Table 4. Description of Installation Prompts . . . . .	51
Table 5. Description of Installation Prompts . . . . .	77
Table 6. Commands to Which Command Access Control Applies . . . . .	142
Table 7. Command Access Quick Reference . . . . .	142
Table 8. Commands to which volume access control applies . . . . .	146
Table 9. Volume Access Quick Reference . . . . .	146
Table 10. Commands for Setting Volume Ownership . . . . .	147
Table 11. Commands and Valid Ownership Keywords . . . . .	150
Table 12. How the Scope of an Audit Affects the Audit Interval . . . . .	164
Table 13. Average Audit Times for Supported LSMs . . . . .	164
Table 14. User-Defined Mixed-Media Files . . . . .	172
Table 15. System-Defined Mixed-Media Files . . . . .	173
Table 16. How ACSLS Uses Mixed-Media Files . . . . .	173
Table 17. CAP Types . . . . .	187
Table 18. CAP States . . . . .	188
Table 19. CAP Modes . . . . .	190
Table 20. Cap Priorities . . . . .	191
Table 21. Cartridge Enter Commands . . . . .	193
Table 22. Archive and Logical Logs . . . . .	215
Table 23. ACSLS Component Types and Identifiers . . . . .	291
Table 24. Valid Components for Audit . . . . .	295
Table 25. Valid Components for Clear Lock . . . . .	304
Table 26. Valid Components for Lock . . . . .	324
Table 27. Valid Lock Types for query lock . . . . .	347
Table 28. Valid Components for Unlock . . . . .	380
Table 29. Valid Components for Vary . . . . .	382
Table 30. Vary Offline Results . . . . .	383
Table 31. Vary offline force Results . . . . .	384
Table 32. Vary Online Results . . . . .	384
Table 33. Addressing Descriptions . . . . .	448
Table 34. Software Drive Numbering . . . . .	452
Table 35. Physical Drive Numbering . . . . .	452
Table 36. CSCI Server Subsystem Environment Variables . . . . .	558

# Preface

---

## ■ About this Book

Automated Cartridge System Library Software (ACSL) is StorageTek's UNIX server software that controls a StorageTek Automated Cartridge System (ACS). The StorageTek ACS family of products consists of fully automated, tape cartridge-based data storage and retrieval systems. ACSL supports network access to different client systems that can range from workstations to mainframes to supercomputers running on a variety of operating systems.

## Audience

This book is for the ACSL system administrator, who uses ACSL to manage a StorageTek ACS. As ACSL system administrator, you should already know the following:

- UNIX file and directory structure
- How to use UNIX commands and utilities for your platform
- UNIX system files
- How to do typical UNIX system administrator tasks, such as logging on as root and setting up user accesses to a UNIX application

## About the Software

This book supports ACSL.

**Note:** The StorageTek Customer Resource Center (CRC) on the World Wide Web lets you download ACSL PTFs and software support for product enhancements such as new drive or library types.

### To access ACSL PTFs and software support for product enhancements:

1. Using an Internet browser such as Netscape, go to the StorageTek CRC. The URL is:  
The CRC is at <http://www.support.storagetek.com/>
2. Select the Request a Login and Password link.
3. Fill in the information requested in the form.

You should receive your account ID and password within two days.

4. When you receive your account information, go back to the CRC and select the `login` link.

When prompted, fill in your User ID and password.

5. After you are logged in, select

Current Products

Select a Product Family: Software

Select ACSLS

(You may want to “View All” to see all the maintenance or documentation.)

## Conventions for Reader Usability

Conventions are used to shorten and clarify explanations and examples within this book.

### Typographic

The following typographical conventions are used in this book:

- **Bold** is used to introduce new or unfamiliar terminology, or it's used in steps to indicate either an action or a decision the user has to make.
- *Letter Gothic* is used to indicate command names, filenames, and literal output by the computer.
- *Letter Gothic Bold* is used to indicate literal input to the computer.
- *Letter Gothic Italic* is used to indicate that you must substitute the actual value for a command parameter. In the following example, you would substitute your name for the “username” parameter.

```
Logon username
```

- A bar ( | ) is used to separate alternative parameter values. In the example shown below either username or systemname must be entered.

```
Logon username | systemname
```

- Brackets [ ] are used to indicate that a command parameter is optional.
- Ellipses ( ... ) are used to indicate that a command may be repeated multiple times.
- This guide shows all ACSLS commands in lowercase. You can, however, enter these commands in all lowercase, all uppercase, or any combination of uppercase and lowercase. Single underlines show minimum command abbreviations. For example, `aud` and `au` are valid forms of the audit command.

## Keys

Single keystrokes are represented by double brackets `[[ ]]` surrounding the key name. For example, press `[[ESC]]` indicates that you should press only the escape key.

Combined keystrokes use double brackets and the plus sign (+). The double brackets surround the key name and the plus sign is used to add the second keystroke. For example, press `[[ALT]]+C` indicates that you should press the alternate key and the C key simultaneously.

## Enter Command

The instruction to “press the <Enter> key” is omitted from most examples, definitions, and explanations in this book.

For example, if the instructions asked you to “enter” `Logon pat`, you would type in `Logon pat` *and* press <Enter>.

However, if the instructions asked you to “type” `Logon pat`, you would type in `Logon pat` and you would *not* press <Enter>.

## Symbols

The following symbols are used to highlight text in this book.

**WARNING:** Information necessary to keep you from damaging your hardware or software.

**CAUTION:** Information necessary to keep you from corrupting your data.

**Hint:** Information that can be used to shorten or simplify your task or they may simply be used as a reminder.

**Note:** Information that may be of special interest to you. Notes are also used to point out exceptions to rules or procedures.

## Technical Support

Refer to the *Requesting Help from Software Support* for information about contacting StorageTek for technical support and for requesting changes to software products. This manual is included the ACSLS program packing in hard copy only.

## Related Documentation

### ACSLs Documentation

The following publications provide more information about ACSLS. They are provided in PDF format on a documentation CD-ROM, and are updated and

maintained on the StorageTek Customer Resource Center (CRC). You can always find the most current documentation updates there.

The *ACSLs Documentation CD-ROM*, is automatically shipped with the product package and provides in PDF format, all the ACSLS documents. These documents are:

- *ACSLs Product Information*
- *ACSLs Installation, Configuration, and Administration Guide*
- *ACSLs Messages*
- *ASLS Release Notes*

## **ACSLs Information on the StorageTek CRC**

In addition to the PDF collections on the *ACSLs Information CD-ROM*, the StorageTek CRC provides PDF collections for ACSLS. Use the following procedure to access this collection on the StorageTek CRC.

### **To access ACSLS PDF collections on the StorageTek CRC:**

1. Using an Internet browser such as Netscape, go to the StorageTek CRC. The URL is:

`http://www.support.storagetek.com/`

2. Select the Request a Login and Password link.
3. Fill in the information requested in the form.

You should receive your account ID and password within two days.

4. When you receive your account information, go back to the CRC and select the Login link.

When prompted, fill in your User ID and password.

5. After you are logged in, select
  - Current Products
  - Select a Product Family: Software
  - Select ACSLS
  - (You may want to "View All" to see all the maintenance or documentation.)

## **ACS Tape Hardware Info on the StorageTek CRC**

The StorageTek CRC provides PDF file format of many of StorageTek's ACS tape hardware publications. Use the following procedure to access these publications on the StorageTek CRC.

### **To access StorageTek ACS tape hardware documentation on the StorageTek CRC:**

1. Using an Internet browser such as Netscape, go to the StorageTek CRC.  
The URL is:

`http://www.support.storagetek.com/`

2. Select the Request a Login and Password link.

3. Fill in the information requested in the form.

You should receive your account ID and password within two days.

4. When you receive your account information, go back to the CRC and select the login link.

When prompted, fill in your User ID and password.

5. After you are logged in, select

Current Products

Select a Product Family: Tape

Select the Tape and Library product documentation you want.



# Customer Contacts

---

## ■ Customer Support

Customer support is available 24 hours a day, seven days a week, to customers with Sun or StorageTek maintenance contracts and to Sun employees. You can find additional information about customer support on the Customer Resource Center (CRC) Web site at:

<http://www.support.storageitek.com>

## ■ Customer-initiated Maintenance

Customer-initiated maintenance begins with a telephone call from you to Sun Microsystems StorageTek Support. You receive immediate attention from qualified Sun personnel, who record problem information and respond with the appropriate level of support.

To contact Sun Microsystems StorageTek Support about a problem:

1. Use the telephone and call:

**☎ 800.525.0369** (inside the United States)

**☎ 303.673.4056** (outside the United States)

2. Describe the problem to the call taker. The call taker will ask several questions and will either route your call to or dispatch a support representative.

If you have the following information when you place a service call, the process will be much easier:

Account name	_____
Site location number	_____
Contact name	_____
Telephone number	_____
Equipment model number	_____
Device address	_____
Device serial number (if known)	_____

Urgency of problem

Fault Symptom Code (FSC)

Problem description

---

---

---

---

---

---

---

---

## ■ Sun's Worldwide Offices

You may contact any of Sun's worldwide offices to discuss complete storage, service, and support solutions for your organization. You can find address and telephone number information on Sun's external Web site at:

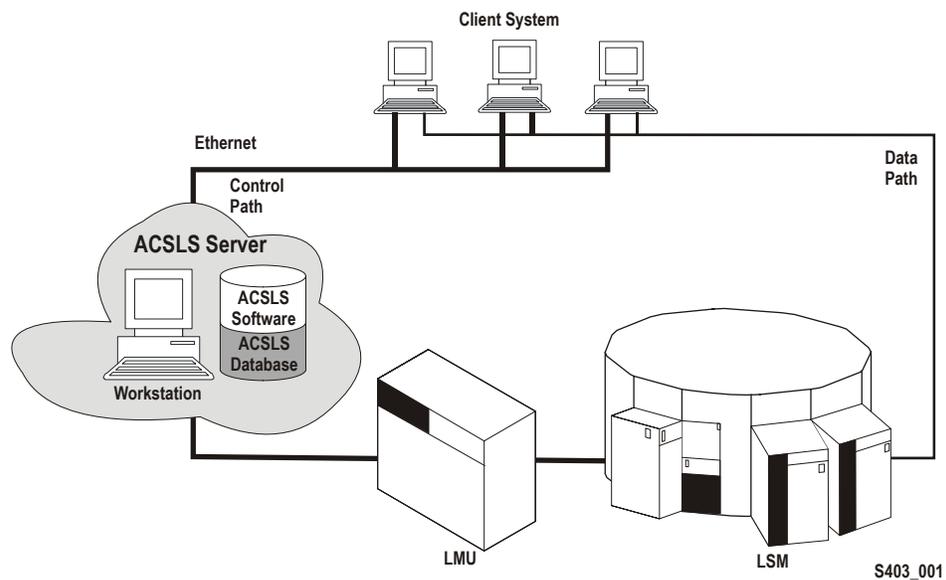
<http://www.sun.com/worldwide/>

## ■ What is ACSLS?

Automated Cartridge System Library Software (ACSL) is StorageTek's server software that controls a StorageTek tape library. An Automated Cartridge System (ACS) is a group of tape libraries connected through pass-thru-ports (PTPs). ACSLS accesses and manages information stored in one or more ACSs through command processing across a network. The software includes a system administration component and interfaces to client system applications, and library management facilities.

**Example:** Figure 1 shows how ACSLS connects the client system with the library. The client system consists of a Client System Component (CSC), an interface between client applications and ACSLS that is written using the CSC developer's toolkit. Independent software vendors commonly write CSCs for their applications.

**Figure 1. Library with ACSLS Server**



## ■ The acssa and acsss User IDs

To control and interact with ACSLS, you use the following user IDs:

`acssa`

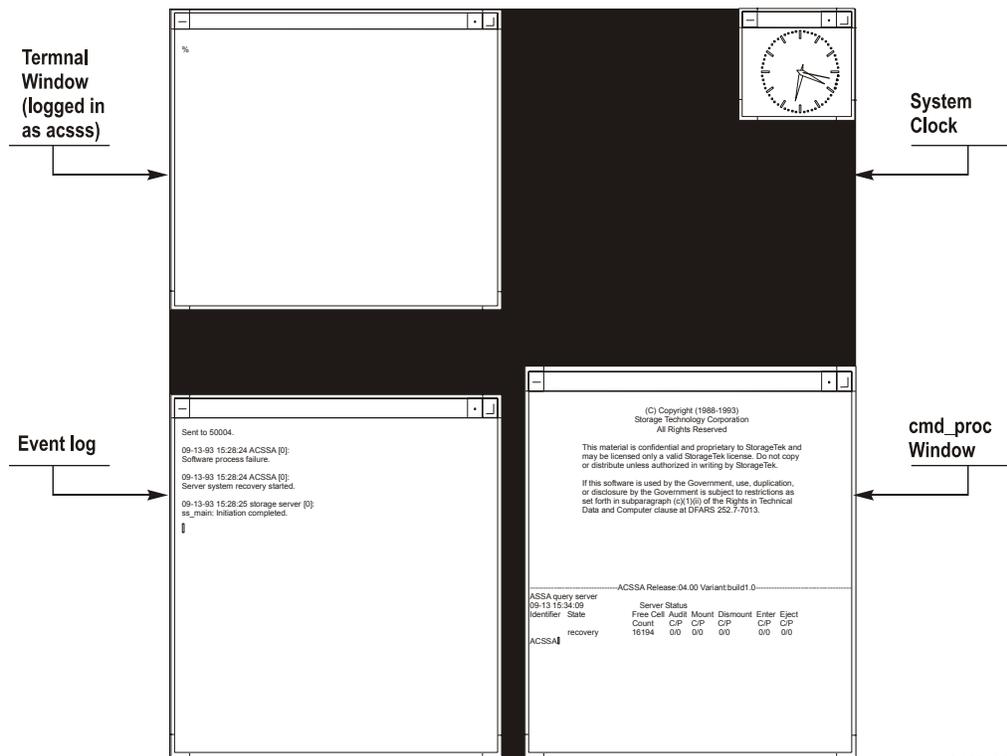
lets you enter ACSLS commands from `cmd_proc`.

`acsss`

lets you run ACSLS utilities from the UNIX command line prompt. You can also open a `cmd_proc` from the `acsss` user ID if you prefer to work from a single user ID; see [“Manually Starting a cmd\\_proc” on page 9](#) for more information.

You can log into multiple windows with both the `acssa` and the `acsss` user IDs. This lets you enter both ACSLS utilities and commands as shown in [Figure 2](#).

**Figure 2. Example Screen Display with multiple windows**



S403\_00:

The screen display shown in [Figure 2](#) contains the following windows:

- **A UNIX terminal window**  
logged in as the `acsss` user.
- **A “tailed” ACSLS Event Log**  
that lets you monitor ACSLS activity; see [“ACSLs Event Log” on page 505](#) for more information.

- **A `cmd_proc` window**  
that lets you enter ACSLS commands.
- **A system clock**  
showing the current date and time.

## ■ Starting ACSLS

You can start ACSLS in either of the following ways:

- Automatically at workstation boot (if the ACSLS server system startup file references the `rc.acsss` utility).

**Hint:** The startup file `/etc/rc` on the ACSLS server contains comments that describe how to modify this file to enable automatic startup.

- Manually by running the `rc.acsss` utility.

When you start ACSLS, the server is in recovery mode. This happens because during its initialization the database and recovery status of the library cartridges and library hardware is checked.

### To manually start ACSLS, do the following:

1. Open a UNIX terminal window.
2. If you are on a remote system, to access the ACSLS server, enter the following command:

```
telnet hostname
```

Where *hostname* is the UNIX hostname of the ACSLS server. Otherwise, go to Step 3.

3. Log in as `acsss`.
4. Enter the following command:

```
rc.acsss
```

**Hint:** You usually start ACSLS in the run state. To start ACSLS in the idle state, enter the following command:

```
rc.acsss idle
```

## ■ Stopping ACSLS

Use this procedure to shut down ACSLS and the database. Typically, you would use this procedure before:

- Performing maintenance on the ACSLS server.

- Backing up the entire ACSLS server disk.

**Hint:** Backing up the entire ACSLS server disk is not the same as backing up the ACSLS database using the `bdb.acsss` utility. See [Chapter 11, “Database Backup and Restore”](#) for procedures for backing up and restoring the database.



**CAUTION:** Do not use the following procedure before upgrading to a new release of ACSLS! See the appropriate chapters for specific procedures for doing an upgrade installation.

**To stop ACSLS, do the following:**

1. If you are not already logged in as `acsss` do so now.
2. Bring up a `cmd_proc`.

You can do this either on the `acsss` user ID or you can log in to `acssa`.

3. From the `cmd_proc`, enter the following commands to idle the server and log off from the `cmd_proc`:

```
idle  
logoff
```

4. On the `acsss` user ID, bring up a UNIX terminal window, enter the following command to stop ACSLS:

```
kill.acsss
```

If `kill.acsss` cannot find a process ID file, the following message appears:

```
kill.acsss: PID file (/tmp/acsss.pid) does not exist, library  
server not terminated.
```

If this message appears, do the procedure in [“Manually Killing the ACSLS Process” on page 7](#), then continue with Step 5.

5. From the UNIX terminal window, enter the following command to shut down the database:

```
db_command stop
```

Wait until you receive the database shutdown message.

## ■ Using a `cmd_proc`

### `cmd_proc` Window

[Figure 3](#) shows the `cmd_proc` window displayed when you log in as `acssa`. The `cmd_proc` window is a split screen where the top section is the message

area and the bottom section is the command area. You enter ACSLS commands at the prompt.

ACSLS must be running to accept commands. If you see the message Query: Query failed. Process failure, wait up to a minute and re-issue the query server command.

**Figure 3. cmd\_proc Window**

```

(c) Copyright (2004)
Storage Technology Corporation
All Rights Reserved

This material is confidential and proprietary to StorageTek and
may be licensed only under a valid StorageTek licence. Do not
copy or distribute unless authorized in writing by StorageTek.

If this software is used by the Government, use, duplication, or
disclosure by the Government is subject to restrictions as set
forth in subparagraph (c)(1)(ii) of the Rights in Technical
Data and Computer Software clause at DFARS 2.52.7-7013.

-----ACSLS x.x.x-----
ACSSA> query server
2004-01-23 15:41:42

IdentifierStateFree CellAudit Mount DismountEnter Eject
Count C/P C/P C/P C/P C/P

run 234 0/0 0/0 0/0 0/0 0/0
>ACSSA

```

## How to Suspend and Resume a `cmd_proc`

If you use “[Manually Starting a `cmd\_proc`](#)” on page 9, you can suspend the `cmd_proc` to perform UNIX commands, and then resume the `cmd_proc`.

**Hint:** Note the following:

1. You must start the `cmd_proc` manually.
2. You must use the C, K, or Bourne shell for this procedure.
3. Any in-process requests that you initiated at the `cmd_proc` will continue to completion while the `cmd_proc` is suspended.

**To suspend and resume a `cmd_proc`, do the following:**

1. While running a `cmd_proc`, press `<CTRL>+Z`.
2. The UNIX shell prompt appears.  
Perform whatever UNIX operations you want.
3. To resume the `cmd_proc`, enter the following UNIX command:  
`fg`

## How to Exit a `cmd_proc`

Use this procedure to terminate an interactive `cmd_proc` session.

**To terminate an interactive `cmd_proc` session, do the following:**

1. While running a `cmd_proc`, wait until all in-process activity is complete and the `ACSSS>` prompt has returned.
2. To exit the `cmd_proc`, enter the following command:

```
logoff
```

The `cmd_proc` terminates.



**WARNING:** Do not “kill” a `cmd_proc` process or exit the session by closing the shell window. Terminating the `cmd_proc` without the proper “logoff” may lock up a shared memory segment which is used by other ACSLS processes.

## Manually Killing the ACSLS Process

If `kill.acsss` cannot find a process ID file, you must manually kill the ACSLS process.

**Note:** To verify if ACSLS is running, enter:

```
psacs
```

**To manually kill the ACSLS process, do the following:**

1. From `cmd_proc`, enter the following command:

```
kill.acsss
```

2. If the `acsss_daemon` and other processes remain in the process table, you can kill the daemon and its children with the command:

```
kill 'pid acsss_daemon'
```

## Starting a cmd\_proc

You can start a `cmd_proc` from any terminal type with a minimum 24 x 80 display size. The terminal type must be defined in the workstation's `/etc/termcap` file.

After logging in, you can start additional `cmd_procs` by entering the `cmd_proc` command from a UNIX terminal window as described in [“Manually Starting a cmd\\_proc” on page 9](#). Typically, you will run only one `cmd_proc`, but you can run as many as your system resources will allow.

**Hint:** A `cmd_proc` session runs independent of ACSLS. If you start a `cmd_proc` session without starting ACSLS, there will be no response to your commands. You may see a socket communication error in the `cmd_proc` if you attempt to run commands while ACSLS is not running. You need not re-start this `cmd_proc` session. It will respond normally once ACSLS is up and running.

## Logging in as acsss from the ACSLS Server

**To start a `cmd_proc` when logged in as `acsss` on the ACSLS server, do the following:**

1. From the ACSLS server, open a UNIX terminal window.
2. To log in as `acsss`, enter the following command:

```
login acsss
```

3. To start the `cmd_proc`, enter the following command:

```
cmd_proc
```

4. Respond to the prompt by entering your terminal type.  
Example of terminal types are `sun`, `sun-cmd`, `xterm`, and `dtterm`.
5. Press `[[RETURN]]`.  
The `cmd_proc` window appears.

## Logging in Remotely as `acssa`

**To start a `cmd_proc` by logging in remotely as `acssa`, do the following:**

1. From a UNIX host on the network, open a UNIX terminal window.
2. To access the ACSLS server, enter the following command:  

```
rlogin hostname -l acssa
```

  
Where *hostname* is the UNIX hostname of the ACSLS server.  
Example of terminal types are `sun`, `sun-cmd`, `xterm`, and `dtterm`.
3. Press `[[RETURN]]`.  
The `cmd_proc` window appears.

## Logging in as `acssa` from a Non-UNIX Client

Use this procedure to start an interactive command `cmd_proc` from a non-UNIX client on the network.

**Hint:** To use this procedure you must have a TCP/IP link to the ACSLS server. Depending on your installation, you may be directly connected to TCP/IP, or you may have to access it manually.

**To start a `cmd_proc` from a non-UNIX client:**

1. If necessary, access TCP/IP.  
See your system administrator for instructions on how to access TCP/IP at your location.
2. To access the ACSLS server, enter the following command:  

```
telnet hostname
```

  
Where *hostname* is the UNIX hostname of the ACSLS server.
3. Log in as `acssa`.
4. Respond to the prompt by entering your terminal type.  
Example of terminal types are `sun`, `sun-cmd`, `xterm`, and `dtterm`.

5. Press `[[RETURN]]`.

The `cmd_proc` window appears.

## Manually Starting a cmd\_proc

Typically, you will manually start a `cmd_proc` from the `acsss` user ID if you do not want to log in as `acssa` to bring up a `cmd_proc`.

**To manually start a `cmd_proc`, do the following:**

1. While logged in as `acssa` or `acsss`, open a UNIX terminal window.
2. To start the `cmd_proc`, enter the following command:

```
cmd_proc
```

3. Respond to the prompt by entering your terminal type.

Example of terminal types are `sun`, `sun-cmd`, `xterm`, and `dtterm`.

4. Press `[[RETURN]]`.

The `cmd_proc` window appears.

## cmd\_proc options

When you manually start a `cmd_proc`, you can also enter the following options:

`-q`

suppresses the automatic `query server` command.

`-l`

brings up `cmd_proc` in command line mode (command area only, no split screen, no message area).

## cmd\_proc Keyboard Shortcuts

[Table 1](#) describes the `cmd_proc` keyboard shortcuts, which are <CTRL>+ keystroke combinations.

**Table 1. cmd\_proc Keyboard Shortcuts**

Key Combination	Action	Notes
<CTRL>+C	Cancels the last <code>cmd_proc</code> command.	<CTRL>+C is the keyboard shortcut for the <code>cancel</code> command. See <a href="#">“cancel” on page 300</a> for more information about the <code>cancel</code> command.
<CTRL>+D	Returns to the <code>cmd_proc</code> prompt.	<CTRL>+D has no effect if the current command has completed. If the current command is processing, it completes but <code>cmd_proc</code> does not display a response message. If you have not entered the current command at the ACSSS prompt, <CTRL>+D deletes the command.
<CTRL>+H	Deletes the previous character on the command line.	On most keyboards, you can also use the <code>[[DELETE]]</code> or <code>[[BACK SPACE]]</code> key.
<CTRL>+I	Refreshes the <code>cmd_proc</code> display	This function is useful if the current <code>cmd_proc</code> display has been corrupted by noise on the communications lines.
<CTRL>+R	Refreshes the current command line.	This function is useful if the current command line display has been corrupted by noise on the communications lines.
<CTRL>+U	Deletes the current command line.	
<CTRL>+Z	Suspends <code>cmd_proc</code> and escapes to the shell environment.	Enter the C shell <code>fg</code> command to resume <code>cmd_proc</code> .

## Redirecting cmd\_proc Inputs and Outputs

You can use an input file to automatically enter commands when you start a cmd\_proc. For example, the following input file verifies ACSLS by mounting and dismounting a cartridge.

```
query drive 0,0,0,0
query cartridge JB1400
mount JB1400 0,0,0,0
dismount JPB1400 0,0,0,0 force
logoff
```

**To start an additional cmd\_proc and specify an input file, do the following:**

1. To start the cmd\_proc, enter the following command:

```
cmd_proc -q < filename
```

You can also start a cmd\_proc, specify an input file, and redirect the output to another file. Using input and output files lets you run a set of commands at cmd\_proc startup and look at the results. For example, the following file shows the results of the commands run in the previous example that showed cmd\_proc with only an input file.

```
ACSSA> query drive 0,0,0,0
1998-06-30 18:23:08
IdentifierStateStatus      Cartridge      Type
0,0,0,0online available          9840
ACSSA> query cartridge JPL1400
1998-06-30 18:23:09
IdentifierStatusCurrent location
JB1400home      0,0,3,0,0
ACSSA> mount JPL1400 0,0,0,0
ACSSA> Mount: JB1400 mounted on 0,0,0,0
ACSSA> dismount JPL1400 0,0,0,0 force
ACSSA> Dismount: Forced dismount of JB1400 from 0,0,0,0
ACSSA> logoff
ACSSA>
```

**To start an additional `cmd_proc`, specify an input file, and redirect the output, do the following:**

1. While logged in as `acssa` or `acsss`, open a UNIX terminal window.
2. To start the `cmd_proc`, enter the following command:

```
cmd_proc -q < file1 > file2
```

Where *file1* is the input file and *file2* is the file to which the output is directed.

By default, `cmd_proc` display area messages are written to `stderr`. but you can also redirect these messages. For example:

```
cmd_proc -q < file1 > file2 2>&1
```

## Idling ACSLS

Use this procedure to suspend request processing by putting ACSLS in the idle state. Typically, this procedure is used before shutting down ACSLS, but you can also use it to temporarily stop ACSLS request processing.

**To idle ACSLS, do the following:**

1. From a `cmd_proc`, enter the following command:

```
idle
```

ACSLs processes all current requests, rejects all new requests, and goes into the idle state.

## Restarting ACSLS

Use this procedure to resume request processing by putting ACSLS in the run state. Typically, you restart ACSLS to remove it from the idle state.

**To restart ACSLS, do the following:**

1. From a `cmd_proc`, enter the following command:

```
start
```

ACSLs resumes request processing.

## ■ ACSLS Directory Structure

Table 2 shows a listing of the directories, subdirectories, and most common used files and shell scripts in ACSLS directory structure.

Two variables are used for ACSLS paths. They are:

- `$ACS_HOME`

This is the home directory for the `acsss` user ID. It is also where the ACSLS product is installed.

- For Solaris, `$ACS_HOME` is `/export/home/ACSSS` by default. However, the parent directory of `$ACS_HOME` is the ACSLS installation directory (by default `/export/home`), which is customer-defined during ACSLS installation on Solaris.

- `$INFORMIX_BACKUP_DIRECTORY`

This is the directory where the ACSLS backups are saved. When second disk support is installed, the `$INFORMIX_BACKUP_DIRECTORY` is moved to the second disk. When second disk support is uninstalled it returns to the primary disk.

- For single disk systems, the default is `/export/backup`
- For second disk systems, the default is `/second_disk/backup`

**Table 2. ACSLS Directory Structure**

Directory	Contents
<code>\$INFORMIX_BACKUP_DIRECTORY</code> (by default <code>/export/backup/</code> )	Database backups (for single disk systems).
<code>/etc/</code>	Miscellaneous UNIX files.
<code>/etc/hosts</code>	List of machines on the network and their addresses.
<code>/etc/passwd</code>	Basic information for each authorized user on the network.
<code>/etc/networks</code>	Information describing the TCP/IP network.
<code>/etc/netmasks</code>	Network masks used to implement IP standard subnetting.
<code>/etc/rc</code>	Command script; includes instructions to start up ACSLS automatically on system reboot.

**Table 2. ACSLS Directory Structure**

<b>Directory</b>	<b>Contents</b>
\$ACS_HOME (by default /export/home/ACSSS/)	Home directory for the <code>acsss</code> user ID. Also the ACSLS home directory. (By default, the <code>ACS_HOME</code> environment variable points to this directory.)
\$ACS_HOME/.acsss_env	Defines the ACSLS environment variables.
\$ACS_HOME/.login	Defines the ACSLS working environment; used by the C shell.
\$ACS_HOME/.profile	Defines the ACSLS working environment; used by the k or Bourne shell.
\$ACS_HOME/kill.acsss	ACSLs shutdown script.
\$ACS_HOME/rc.acsss	ACSLs startup script.
\$ACS_HOME/bin/ (by default /export/home/ACSSS/ bin/)	Contains executable shell scripts and programs.
\$ACS_HOME/bin/ acsss_config	ACSLs configuration program.
\$ACS_HOME/bin/ backup.sh	Automatic database backup script.
\$ACS_HOME/bin/ bdb.acsss	Manual database backup script.
\$ACS_HOME/bin/ db_command	Database startup/shutdown script.
\$ACS_HOME/bin/ db_export.sh	Database export script.
\$ACS_HOME/bin/ db_import.sh	Database import script.
\$ACS_HOME/bin/ rdb.acsss	Database recovery script.
\$ACS_HOME/bin/ volrpt.exe	Executable to run cartridge report script.
\$ACS_HOME/config/ (by default /export/home/ACSSS/ config/)	Contains ACSLS configuration files.

**Table 2. ACSLS Directory Structure**

<b>Directory</b>	<b>Contents</b>
\$ACS_HOME/data/ external/  (by default /export/home/ACSSS/ data/external)	Contains customized files used in access control, mixed media, and cartridge reporting.
\$ACS_HOME/data/ external/ access_control/	Contains access control sample and customized files.
\$ACS_HOME/data/ external/ access_control/ adi.names	Lists ADI names and user IDs of all hosts (OSLAN protocol).
\$ACS_HOME/data/ external/ access_control/ adi.names.SAMPLE	Sample adi.names file; <u>not</u> actually used for controlling access.
\$ACS_HOME/data/ external/ access_control/ command.ALL.allow	Lists users allowed to use all commands.
\$ACS_HOME/data/ external/ access_control/ command.command.allow	Lists users allowed to use this command.
\$ACS_HOME/data/ external/ access_control/ command.command. disallow	Lists users not allowed to use this command.
\$ACS_HOME/data/ external/ access_control/ command.SAMPLE.allow	Sample command allow file; <u>not</u> actually used for controlling access.
\$ACS_HOME/data/ external/ access_control/ command.SAMPLE. disallow	Sample command disallow file; <u>not</u> actually used for controlling access.

**Table 2. ACSLS Directory Structure**

<b>Directory</b>	<b>Contents</b>
\$ACS_HOME/data/ external/ access_control/ internet.addresses	Lists Internet addresses and user IDs of all hosts (TCP/IP protocol).
\$ACS_HOME/data/ external/ access_control/ internet.addresses. SAMPLE	Sample internet.addresses file; <u>not</u> actually used for controlling access.
\$ACS_HOME/data/ external/ access_control/ ownership.assignments	Information about assigning ownership to cartridges.
\$ACS_HOME/data/ external/ access_control/ users.ALL.allow	Lists users allowed to access owned cartridges with all commands.
\$ACS_HOME/data/ external/ access_control/ users.ALL.disallow	Lists users not allowed to access owned cartridges with any command.
\$ACS_HOME/data/ external/ access_control/ users. <i>command</i> .allow	Lists users allowed to access owned cartridges with this command.
\$ACS_HOME/data/ external/ access_control/ users. <i>command</i> . disallow	Lists users not allowed to access owned cartridges with this command.
\$ACS_HOME/data/ external/ access_control/ users.SAMPLE.allow	Sample cartridge allow file; not actually used for controlling access.
\$ACS_HOME/data/ external/ access_control/ users.SAMPLE.disallow	Sample cartridge disallow file; not actually used for controlling access.
\$ACS_HOME/data/ external/ fixed_cartridge	Contains Extended Store Feature sample and customized files.

**Table 2. ACSLS Directory Structure**

<b>Directory</b>	<b>Contents</b>
\$ACS_HOME/data/ external/ fixed_cartridge lsm_fixed_cartridge	Extended Store Feature Control file; must be modified to enable this feature.
\$ACS_HOME/data/ external/mixed_media/	Contains mixed-media sample and customized files.
\$ACS_HOME/data/ external/mixed_media/ scratch_preferences. dat	Lists transport types and a (user-defined) prioritized list of media types to use when selecting scratch cartridges.
\$ACS_HOME/data/ external/mixed_media/ scratch_preferences. SAMPLE	Sample scratch preferences file; <b>not</b> actually used for controlling media.
\$ACS_HOME/data/ external/volrpt/	Contains cartridge report sample and customized scripts.
\$ACS_HOME/data/ external/volrpt/ owner_id.volrpt	Sample custom cartridge report input file.
\$ACS_HOME/data/ internal/  (by default /export/home/ACSSS/ data/internal/)	ACSLs internal files. Please do not modify.
\$ACS_HOME/data/ internal/mixed_media/	Contains ACSLS internal mixed media files. Do not modify these files.
\$ACS_HOME/data/ internal/mixed_media/ drive_types.dat	Lists supported transport types.
\$ACS_HOME/data/ internal/mixed_media/ media_cleaning.dat	Lists transport types and compatible cleaning cartridge types.
\$ACS_HOME/data/ internal/mixed_media/ media_compatibility. dat	Lists transport types and compatible media types.
\$ACS_HOME/data/ internal/mixed_media/ media_types.dat	Lists supported media types.

**Table 2. ACSLS Directory Structure**

<b>Directory</b>	<b>Contents</b>
\$ACS_HOME/diag/	Contains diagnostic files and shell scripts.
\$ACS_HOME/diag/bin/ pinglmu.sh	Post-configuration tool to check connections between ACSLS server and serial-attached libraries.
\$ACS_HOME/diag/bin/ probe-scsi.sh	Post-configuration tool to check connections between ACSLS server and SCSI libraries.
\$ACS_HOME/diag/bin/ README.txt	Describes the utilities in /export/home/ACSSS/diag/bin.
\$ACS_HOME/diag/data/ ivp.dat	Data file used to drive the IVP mount/dismount activities.
\$ACS_HOME/diag/ivp/ ivp.sh	Initial Verification Program (IVP) used for testing the mount/dismount commands.
\$ACS_HOME/diag/ivp/ README.ivp	Explanation of the IVP process.
\$ACS_HOME/install/	Contains ACSLS installation scripts.
\$ACS_HOME/install/ full_disk.sh	Database automatic checkpoint backup script.
\$ACS_HOME/install/ sd_mgr.sh	Second disk installation script.
\$ACS_HOME/lib/ <b>(by default</b> /export/home/ACSSS/ lib/)	Contains ACSLS installed shared libraries required at runtime.
\$ACS_HOME/log/ <b>(by default</b> /export/home/ACSSS/ log/)	Contains ACSLS event log and utility event log files.
\$ACS_HOME/log/ acsss_config.log	Report of the library hardware configured by the acsss_config program.
\$ACS_HOME/log/ acsss_config_event. log	Logs events for the acsss_config program.
\$ACS_HOME/log/ acsss_event.log	ACSLs event log.
\$ACS_HOME/log/ bdb_event.log	Logs events for the bdb_acsss script.

**Table 2. ACSLS Directory Structure**

<b>Directory</b>	<b>Contents</b>
<code>\$ACS_HOME/log/cron_event.log</code>	Logs events for the cron job that calls the <code>full_disk.sh</code> script.
<code>\$ACS_HOME/log/export_event.log</code>	Logs events for the <code>db_export.sh</code> script.
<code>\$ACS_HOME/log/full_disk.sh.log</code>	Logs events for the <code>full_disk.sh</code> script.
<code>\$ACS_HOME/log/import_event.log</code>	Logs events for the <code>db_import.sh</code> script.
<code>\$ACS_HOME/log/install.log</code>	Logs events for the <code>install.sh</code> script.
<code>\$ACS_HOME/log/rdb_event.log</code>	Logs events for the <code>rdb.acsss</code> script.
<code>\$ACS_HOME/log/scsilh.log</code>	Logs events for SCSI LSM activity while in diagnostic mode.
<code>\$ACS_HOME/log/sd_event.log</code>	Logs events for the <code>sd_mgr.sh</code> script.
<code>\$ACS_HOME/log/volrpt.log</code>	Logs events for the cartridge report script.
<code>\$BASEDIR/ACSSA/</code> (by default <code>/export/home/ACSSA/</code> )	<code>acssa</code> home directory.
<code>\$BASEDIR/informix/</code> (by default <code>/export/home/informix/</code> )	Database home directory.
<code>/second_disk/</code>	Second disk files
<code>\$INFORMIX_BACKUP_DIRECTORY</code> (by default <code>/second_disk/backup/</code> )	Database backups when second disk support is installed.
<code>/tmp/</code>	Temporary files.
<code>/tmp/ acsss.pid</code>	ACSL S process ID file; used in ACSLS startup and shutdown.



## ■ ACSLS License Key Utility

A license key limits the number of cartridges in your library, not the number of slots. Even if you add additional libraries to your configuration you do not need a new license key until you need to add additional cartridges.

**Example:** If you have a license key for 6000 cartridges and have purchased an additional library, you can move the 6000 cartridges between the libraries and not need a new license key. However, if you add additional cartridges, you will need a new license key.

After you install your license key, you have a 75-day trial period. Before your 75-day period expires, you must secure a permanent license key. Do not wait until this trial period expires to secure your permanent key.

Keep in mind that you have a full 75-day trial period once you have installed ACSLS. When you have 60 days left on your trial period, and if you have not entered your permanent license key, ACSLS will begin displaying warning messages notifying you of the date when your trial period will expire. You must request and enter your permanent license key before it expires.

Once the trial period key expires, ACSLS automatically terminates, and you cannot re-start ACSLS.

The `licensekey.sh` utility activates the license key that allows you to use ACSLS. To run the license key utility script, enter:

```
licensekey.sh
```

The two types of license keys (excluding the 75-day trial period) are:

- **Permanent:** Enables you to use ACSLS when StorageTek has received the initial license fee for a product or feature.
- **Emergency Software Key:** For short-term use in emergency situations. It is limited to only seven (7) days. Do not panic and overlay your 75-day trial period with a 7 day emergency license key.

**Note:** After the trial period expires and a valid license key has not been entered or an emergency license key expires, ACSLS automatically shuts down and will not re-start.

## Obtaining a Permanent Key

1. Access the StorageTek Customer Resource Center (CRC) at the following URL: <http://www.support.storagetek.com/>
2. Click Tools & Services
3. Click Software Keys.
4. Scroll to ACSLS and click ACSLS x.x.x Key Request.

The ACSLS x.x.x Key Order Form displays.

5. Click your platform type.
6. Click permanent.

**Note:** You must furnish the following information:

- The StorageTek SAP System Order Number. (This number appears on the packing list shipped with the ACSLS Software.)
- Your Customer Name as it appears on your StorageTek Software License Agreement.
- Name, telephone number, and e-mail address of the person within your company who is to receive the key.
- Your Customer Site ID.
- Number of cartridges for which you are licensed.

Keys are issued within 48 hours of the license key request, Monday through Friday, 7:00 a.m. to 4:00 p.m. Mountain Time, except holidays.

## Obtaining an Emergency Key

If an emergency (temporary) key is required outside of the business hours listed above:

1. Access the StorageTek Customer Resource Center (CRC) at the following URL: <http://www.support.storagetek.com/>
2. Click Tools & Services
3. Click Emergency Software Keys.

The StorageTek Emergency Generator screen displays.

All fields with an asterisk are required and must be completed before submitting the license key request.

## License Key Format

License key parameters must be entered **exactly** as received from StorageTek.

### Example:

```
licensekey.sh CUSTOMER_NAME 097531 2003057 ACSLSxxx
FWZZCH*DYZUP383Q
```

**Note:** If the customer name consists of more than one word (that is, it contains embedded blanks), the customer name must be enclosed in double quotation marks.

**Hint:** Copy and paste from your license key email to avoid typing errors.

An ACSLS license key consists of the following fields:

1. The customer name (64-character maximum, valid characters include A-Z, 1-9, \*, and blank). All letters must be upper case.

**Note:** If the customer name consists of more than one word (that is, it contains embedded blanks), the customer name must be enclosed in double quotation marks.

*CUSTOMER\_NAME* is the customer name.

2. The site number, consisting of from four to six numeric characters.

*nnnnnn* is the site number.

3. The expiration date, expressed in seven numeric characters (in ISO standard date format).

*yyyyddd* is the license key expiration date.

**Hint:** The license key expiration date is expressed:

- *yyyy* = year expressed in four numerals, such as 2003.
- *ddd* = the number of days into the year counting from January 1 (the Julian day of the year), such as, 229 is the two-hundred twenty-ninth day of the year.

4. A product identification number, consisting of eight alphanumeric characters

*product\_identifier* is composed of the product name (ACSLs) followed by the three (3) character release ID, such as 7.0.0.

5. An encoded license key between 14 and 17 characters long.

*license\_key\_string* is the encrypted license key option string.

## License Key Validation

ACSLS validates the license key: during product initialization; at a preset time each day; before cartridges are entered; and after audits complete. ACSLS shuts down when the 75-day trial period ends or the emergency license key expires. After the trial period, the product will not initialize at system startup if you do not have a valid ACSLS license key. While the ACSLS system is up and running, warning messages are issued as you approach your licensed library capacity, or as you approach the trial period or license key expiration dates.

Your ACSLS license supports a maximum number of cartridges. When you exceed your licensed cartridge limit, you cannot enter additional cartridges. If you need to increase your library capacity, you must upgrade your ACSLS license to support the additional cartridges.

Use the `get_license_info` utility to display the number of cells (cartridges) supported by your current license key. The `get_license_info` utility also displays the number of used (allocated) cells, free cells, and total available cells in all libraries managed by ACSLS. Refer to “[get\\_license\\_info](#)” on [page 265](#) for procedures.

When you exceed your licensed cartridge limit, you cannot enter additional cartridges.

## License Key Operational Procedures

License keys are entered, updated, or replaced by invoking the license key script from the command line.

1. To initialize ACSLS with a valid license key,
  - After configuring the database with `acsss_config`, but prior to entering `rc.acsss`, run the `licensekey.sh` script.
  - Enter the required fields described above.
 

**Hint:** Copy and paste from your license key email to avoid typing errors.
  - Enter: `rc.acsss`  
ACSLS initializes.
2. To update or replace an ACSLS license key,
  - Run `licensekey.sh` with the correct parameters. This can be done online without recycling ACSLS.  
ACSLS should accept the new license key.
3. If you experience errors with your ACSLS license key,

- Consult the *ACSLs Messages Manual, Version x.x.x*, and follow the instructions for the error message you received.
- If you are still unable to resolve the problem, contact the StorageTek Software Manufacturing and Distribution department (1-800-436-5554) and select option three, or call your StorageTek Marketing Representative or Systems Engineer during normal business hours.



# Installing ACSLS 7.1 on Solaris

## 9

# 3

---

ACSLs 7.1 supports Solaris 9. StorageTek has not tested down level versions of the operating system for this release. This chapter describes procedures for installing ACSLS on Solaris 9 from CDROM.

To apply any maintenance, refer to the installation instructions in the associated PUT or SPE or PTF document included with the CD-ROM, or the document associated with the maintenance tar file on the Customer Resource Center (CRC) web page.

This chapter takes you through and discusses the following steps for installing ACSLS on Solaris 9. In these steps:

1. Have your license key information and ACSLS 7.1 CD-ROM available.
2. Export the database if you are upgrading from a previous version of ACSLS.
3. Prepare for Solaris 9 installation.
4. Install the Solaris 9 operation system.
5. Prepare for ACSLS installation.
6. Install ACSLS.
7. Import the database from any previous version.

**Note:** If you import the database, you can skip the next step.

8. Run `acsss_config` to configure your library hardware (see [“Configuring or Updating your Library Hardware” on page 130](#)).
9. Enter your license key information (see [Chapter 2, “ACSLs License Key”](#)). A 75-day trial license is automatically installed when ACSLS is initially installed.
10. Reset any custom dynamic or static variables. Performing an import and selecting option 4 merges any previous customized dynamic variables.
11. Audit new libraries.
12. Verify the ACSLS installation.

In addition, this chapter discusses:

- ACSLS Setup for Client Communication
- Regressing to a previous version of ACSLS
- Uninstalling ACSLS 7.1

## ■ Step 1: Export the Database

Perform this step if you are upgrading from a previous version of ACSLS (see [“Exporting the Database” on page 206](#)).

## ■ Step 2: Preparing for Solaris Installation

Before installing Solaris, complete the following steps:

1. Have your license key information and ACSLS 7.1 CD-ROM available.
2. Satisfy all requirements specified in *Product Information* (see [“ACSL S Information on the StorageTek CRC” on page xl](#)) prior to installation of Solaris or ACSLS.

ACSL S 7.1 requires a minimum of 4GB on the primary disk system.

3. Install the library hardware according to the Sun StorageTek installation instructions.

**Note:** If you are installing a SCSI library, you should connect the server to the library before installing ACSLS. The `install.sh` script installs device drivers and kernel patches. If the SCSI cable is not connected, the device driver installation fails. However, if library hardware is not available at the time of software installation, the driver can be installed separately at a later time.

4. Install the ACSLS server hardware according to the hardware manufacturer's instructions.
5. Determine and record information about network communication settings.

**CAUTION:** Consult your system administrator for assistance in obtaining this information. It is critical that this information is correct in order for ACSLS to run properly.

**Table 3. Pre-Installation Worksheet**

Installation Options	Your System Values
<input type="checkbox"/> Host Name	
<input type="checkbox"/> Network Adapter	
<input type="checkbox"/> Internet Protocol Address	
<input type="checkbox"/> Name Service	

**Table 3. Pre-Installation Worksheet**

Installation Options	Your System Values
<input type="checkbox"/> Network Mask	
<input type="checkbox"/> Domain Name	
<input type="checkbox"/> Gateway Address	
<input type="checkbox"/> Name Server	
<input type="checkbox"/> Subnets	
<input type="checkbox"/> System Type	

## ■ Step 3: Installing the Solaris 9 Operating System

### To install the Solaris 9 operating system:

1. Boot your system from the CD-ROM containing the Solaris operating system:

Insert the Solaris CD (**Solaris 9 Software**) into the CD-ROM drive. This is CD-1 (of 2).

**Do *not* insert the Solaris 9 Installation CD.** Doing this installs the incorrect version for using ACSLS.

- a. Press `[[STOP]]+A`.
- b. From the `ok` prompt, enter

```
boot cdrom
```

Press `<Enter>`.

**Note:** If you have configured the SCSI/Ethernet card on your server and you are not using the second Ethernet connection, then disregard the message, "home1: link down".

**Hint:** The `Solaris Install Console` will appear first. During the installation process, this console provides messages indicating what the system is doing.

2. Select Language.  
The choices are 0-9. The menu explains each choice.
3. Select Locale options.  
`sysidtool` is started. Press `[[F2]]` twice to continue.
4. In the Network Connectivity window, select `yes` and then `[[F2]]` to continue.

5. In the DHCP window, unless you are using DHCP, select no and [[F2]] to continue
6. In the Primary Network Interface window, enter the device name and press [[F2]] to continue.
7. In the Host Name window, enter the Host Name for your system and press [[F2]] to continue.
8. Enter your IP address in the IP Address window and press [[F2]] to continue.
9. Select yes in the Subnet window if the system on a subnet. Enter no if it is a standalone.
10. Press [[F2]] in the Net Mask window if the information is correct.
11. To enable IPv6 select no; press [[F2]].
12. Make a selection in the Default Route window.  
A confirmation window displays notifying you that it is looking for the route. Once it displays the route, press [[F2]] to continue.
13. Confirm the Host Name, Networked, and IP address in the Confirm Information window, and press [[F2]].
14. Select no for Configure Security Policy and press [[F2]].
15. Confirm Kerbous security.
16. In the Primary Network Interface window, enter your device name and then select [[F2]].
17. Select the name service used in your network in the Name Service window, and press [[F2]].

**Notes:**

- Select none
  - Do *not* select the NIS+ or NIS or DNS services unless your system administrator can configure these services correctly.
  - If one of these services is selected, a number of different prompts appear. Respond to these prompts appropriately.
18. Confirm the Name Service in the Confirm Information window, and press [[F2]].
  19. Specify your time zone type in the Geographical Regions window and press [[F2]].
  20. Enter the country and region and press [[F2]].

21. Enter the current date and time (if required) in the Date and Time window and press **[F2]**.
22. Press **[F2]** on the Confirm Information window.  
The Solaris Interactive Installation window displays.  
Select Initial.
23. The Solaris Interactive Installation window displays a message similar to the following.  
There are two ways to install  
The choices are: standard or Flash.
24. Select **[F2]** - standard.
25. Verify your regions setting in the Select Geographic Regions window and press **[F2]**.
26. Select to Include Solaris 64 Bit Support to enable 64-bit support. (This applies to SUN Ultra platforms.).
27. Select the end user and press **[F4]** to customize  
The Customize Software window displays displaying the Software Clusters and Packages window.
28. Scroll down and locate On-Line Manual Pages in the window. Click once to fully select the Man Pages.
29. Continue scrolling down and locate OpenWindows Version 3 in the window.
30. Select 64-bit and press the space bar to fully select the option.
31. Continue to scroll down through the main directory to Programming tools and libraries.
  - a. Select the triangular arrow (Collapsed Cluster) in the left margin of Programming tools and libraries to expand the subdirectory.
  - b. Select CCS tools bundled with SunOS.
  - c. Select the triangular arrow (Collapsed Cluster) in the left margin of Programming tools and libraries to close the subdirectory.
32. Press OK to exit the Customize Software window.
33. Press **[F2]** when the Select Software window displays.
34. If the boot disk is displayed in the Available Disks window, select it, then press the single-right-arrow button to move it to the "Selected disks" column.

Repeat this step for the second disk and move it in the second column.

35. Press **[[F2]]** on the `Select Disks` window.

The `Preserve Data?` window displays.

36. Press **[[F2]]** for a new installation and/or to overwrite current file systems and unnamed file systems. (Press `Preserve . . .` to preserve existing data during an upgrade installation.)

37. Press `Manual Layout` in the `Automatically Layout File Systems?` window and **[[F4]]**.

38. Press `Customize` in the `File System and Disk Layout` window to define disk partitions for ACSLS. Press **[[F4]]** to customize.

The `Customize Disks` window displays.

39. Select the disk to customize and press **[[F4]]**.

How you set up your disk partitions can have a significant impact on system performance. This is because disk I/O is used for file system operations and also by the virtual memory subsystem for paging and swapping.

In the following discussion, `/export/home` is used as the directory where ACSLS is installed; however, in this release ACSLS now installs to any directory in Solaris.

In Solaris, you can configure the backup directory.

- In a single disk installation the default is `/export/backup`
- In a second disk installation the default is `/second_disk/backup`

Enter the values specified to define primary disk partitions for the database:

- Assign the mount point for each of the eight partitions in the left column.
- Assign the size for each slice in the right column except for `/export/backup`.
  - It is suggested that you do *not* make root any smaller than 100 MB.
  - Swap space must be no smaller than 500 MB.
  - The size of `/usr` must be at least 550 MB.
  - The size of `/var` must be at least 250 MB.
  - `/export/home` and `/export/backup` must be at least 1GB.

You may be able to improve disk I/O performance by following the configuration guidelines described in the figures below.

The following figure is a guideline for partitioning a 4 GB disk.

**Figure 4. Primary Disk Partitions Guidelines**

Disk: c0t0d0		Disk Size: <b>4GB</b>
		Minimum Requirements
0	/	<b>100 MB</b>
1	<b>swap</b>	<b>500 MB</b>
2	<b>backup</b> This is your total disk size.	<b>4 GB</b>
3	<b>/var</b>	<b>250 MB</b>
4	<b>/export/home</b>	<b>1GB</b>
5	<b>/opt</b>	<b>100 MB</b>
6	<b>/usr</b>	<b>550 MB</b>
7	<b>/export/backup</b>	The remaining disk memory is assigned to this partition
Capacity		<i>disk size</i>
Allocated:		yyMB or GB
Free:		xxMB or GB

The following figure is a guideline for partitioning a 9+ GB disk.

**Figure 5. Primary Disk Partitions Guidelines**

Disk: c0t0d0		Disk Size: <b>9GB</b>						
		Minimum Requirements						
0	/	<b>100 MB</b>						
1	swap	<b>500 MB</b>						
2	backup This is the total disk size.	<b>9 GB</b>						
3	/var	<b>250 MB</b>						
4	/export/home	<b>1GB</b>						
5	/opt	<b>100 MB</b>						
6	/usr	<b>550 MB</b>						
7	/export/backup	The remaining disk space is assigned to this partition						
<table> <tr> <td>Capacity</td> <td><i>disk size</i></td> </tr> <tr> <td>Allocated:</td> <td>yyMB or GB</td> </tr> <tr> <td>Free:</td> <td>xxMB or GB</td> </tr> </table>			Capacity	<i>disk size</i>	Allocated:	yyMB or GB	Free:	xxMB or GB
Capacity	<i>disk size</i>							
Allocated:	yyMB or GB							
Free:	xxMB or GB							

40. Define a partition called /second\_disk.
41. Press **[[F2]]** to continue.
42. Select the second drive to customize and press **[[F4]]**.
43. Define a partition called /second\_disk.
44. Press **OK** after you have filled out the `Customize Disks` window.  
 The `File System and Disk Layout` window displays.
45. Are the file systems and disk layout correct in the `File System and Disk Layout` window?  

<b>YES</b> Press <b>[[F2]]</b> to accept the layout displayed.
<b>NO</b> Press <code>Customize...</code> to change the file system or layout information.
46. Press **[[F2]]** in the `Mount Remote File Systems?` window.
47. Verify that the `Profile` window displays the configuration that you want and press `Begin Installation`.

48. Select `Auto Reboot` in the next window and press `[[F2]]` to begin the installation.

The installation process takes from 30 minutes to 2 hours depending on the speed of the CD-ROM drive. The system activities can be monitored by following the messages in the `Solaris Install Console` and by the `Installing Solaris Software - Progress` window, which shows the progress with a sliding bar.

49. When the installation is complete, the system reboots and prompts for a password. Reenter the same password at the prompt to verify. These passwords must be maintained or set to never expire.

50. A message similar to the following displays:

```
This system is configured to conserve energy. Do you
want this automatic power saving shutdown?
```

51. Select `n`.

A message similar to the following displays:

```
Do you want this system to ask about this again?
```

52. Select `n`.

53. Specify the media from which you are installing Solaris 9 Software.

Select one of the following:

- `CD`
- `Network File System`

54. Insert **Solaris 9 Software - CD 2 of 2** when prompted and click `OK`.

55. Click `Next`.

The cd is ejected.

56. Click `Reboot Now`.

57. Change directories to the `/etc` directory.

```
cd /etc
```

58. Edit the `hosts` file: add the name of each client and the IP address of each client to the end of the file.

59. At the system prompt enter `df -k` to verify the disk space for `/export/home` and `/export/backup`.

- There should be at least 1 GB disk space for the `/export/home` file system for the database. `/export/home` will contain the contents of the installation media.
- There should be at least 1 GB disk space for `/export/backup`.

## ■ Step 4: Preparing for ACSLS Installation

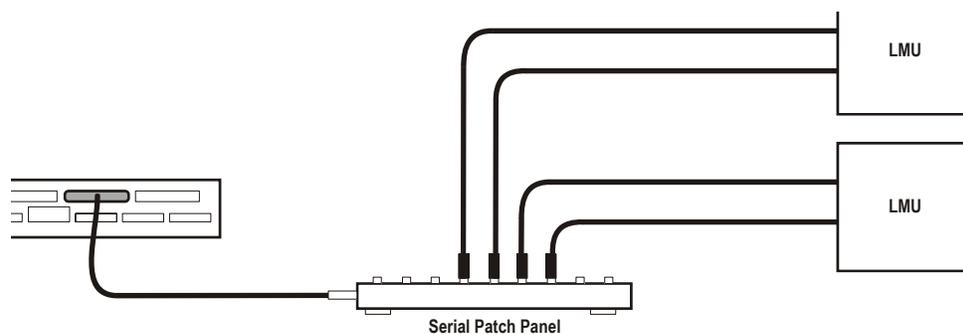
Before you install ACSLS, complete the following steps:

1. Verify that the server system hardware is properly configured, connected, powered on, and ready.
2. Connect the LMU to a valid serial port.

If more than one port is available on your server, configure two ports to the LMU. An alternate LMU connection provides higher throughput and greater robustness.

3. Multiple serial port connects are necessary if you are configuring the server to a dual-LMU installation. You can provide redundant connections to each LMU using a multi-port adapter as suggested in [Figure 6](#).

**Figure 6. Dual LMU Connection Configuration**



S403\_003

4. If your installation includes a SL8500 library or a 9300 library and TCP/IP LMU(s), connect the TCP/IP LMU(s) to the network used for LMU communication.
5. For SCSI-connected libraries, you should use a differential connection where possible. If a single-ended SCSI controller is used, you should limit the cable distance to three meters between the server and the library. With low-voltage differential (LVD), the cable should be no more than 10 meters. High-voltage differential (HVD) SCSI cables can extend up to 20 meters.
6. Make sure that each attached LMU and LSM is fully configured, powered on, and ready.

**Note:** The configuration script, `acsss_config`, will fail unless all LMUs and LSMs are fully configured, powered on, and ready.

7. If you have any communication problems refer to the troubleshooting chapter in the appendix.

8. If you intend to use removable media for database backup, have a blank cartridge available for your backup tape device to complete the configuration process.
9. ACSLS requires specific user IDs. If these user IDs are not defined before ACSLS installation, ACSLS creates them. When ACSLS installation creates the user IDs, the system assigns the user ID numbers and group ID numbers. To assign specific user ID and group ID numbers, you must define the following groups and users before installing ACSLS:

<b>User</b>	<b>Group</b>
acsss	staff
acssa	staff
lib6	staff
informix	informix

When these user IDs are defined before ACSLS installation, they can either be defined locally (on the ACSLS server) or via remote authentication methods (e.g., NIS or Kerberos). These user IDs must be defined with the following properties:

- The shell for these user IDs must be sh on Solaris. The exception is acssa (ksh).
- The home directories for the acsss, acssa, and informix user IDs must be under the ACSLS installation directory. By default, the ACSLS installation directory is /export/home (referred to as \$BASEDIR). The home directories for the ACSLS user IDs are:

acsss	\$BASEDIR/ACSSS
acssa	\$BASEDIR/ACSSA
informix	\$BASEDIR/informix
lib6	\$BASEDIR/ACSSS

If the home directories for the ACSLS user IDs do not match their required locations, please either modify the home directories for these users or delete the user IDs so they are added correctly during the ACSLS installation process.

The following command creates the acsss user on Solaris. (You must be logged in as root.)

```
useradd -d $BASEDIR/ACSSS -g staff -s /bin/sh -c "ACSL Control Login" acsss
```

The account information is:

```
acsss -d $BASEDIR/ACSSS-g staff -c "ACSL Control Login"-s /bin/sh
acssa -d $BASEDIR/ACSSA-g staff -c "ACSL SA Login"-s /bin/ksh
informix -d $BASEDIR/informix/IDS7.3-g informix-c "ACSL Informix Database Owner"-s /bin/sh
lib6 -d $BASEDIR/ACSSS-g staff -s /bin/sh
```

The following commands modify the `acsss`, `acssa`, and `informix` users' home directories. (You must be logged in as `root`.)

```
usermod -d $BASEDIR/ACSSS acsss
usermod -d $BASEDIR/ACSSA acssa
usermod -d $BASEDIR/informix informix
```

## ■ Step 5: Installing ACSLS

### Remove any previous version

1. Have your license key available.
2. Is this a new installation?

---

YES    Go to [“Install using pkgadd:” on page 40](#).

---

NO     Make sure you exported the database by using the `db_export.sh` utility command.

---

3. Shut down ACSLS:

You need to be logged in as `acsss` to do this.

- a. Enter the following:

`idle` (from a `cmd_proc`)

`kill.acsss` (from a command prompt)

- b. Enter one of the following, depending on the ACSLS release from which you are upgrading:

- If you are upgrading from ACSLS 5.3 or 5.3.2:

**`db_command stop`**

- If you are upgrading from ACSLS 5.4:

`db_command.sh stop`

- If you are upgrading from ACSLS 6.0 and later:

**`db_command stop`**

4. Login as `root` and remove ACSLS, backup and other files:

For ACSLS 5.3 or 5.3.2, enter:

- `cd /export/home`
- `rm -rf ACSSS oracle`
- `cd /export/backup`
- `rm -rf oracle misc`

For ACSLS 5.4, enter:

- `cd /export/home`
- `rm -rf ACSSS sybase`
- `cd /export/backup`
- `rm -rf sybase misc`

For ACSLS 6.0 or 6.0.1, enter:

- `cd /export/home`
- `rm -rf ACSSS informix`
- `cd /export/backup`
- `rm -rf informix misc`

If the `server_to_server` directory exists, enter:

```
rm -rf informix misc server_to_server
```

- `cd /`
- `rm -rf INFORMIXTMP nsr`

For ACSLS 6.1.0, 6.1.1, or 7.0 enter:

- `cd /export/home`
- `pkgm STKacsls`
- `rm -rf ACSSS ACSSA informix`
- `cd /export/backup`
- `rm -rf informix misc server_to_server`
- `cd /`
- `rm -rf INFORMIXTMP`
- `rm -rf /nsr`
- `cd /var/tmp`
- `rm -rf acsls`

5. Remove files under second disk (if installed)

- `cd /second_disk`

If you installed the second disk in another directory other than `/second_disk`, `cd` to that directory.

- `rm -rf data`

- `cd /second_disk/backup`

If you installed the second disk in another directory other than `/second_disk/backup`, `cd` to that directory.

For ACSLS 5.3 or 5.3.2, enter:

- **`rm -rf oracle misc`**

For ACSLS 5.4, enter:

- `rm -rf sybase misc`

For ACSLS 6.0 and later enter:

- `rm -rf informix misc`

6. If you want to reinstall ACSLS in a different ACSLS installation directory, either delete the `acsss`, `acssa`, and `informix` user IDs or modify their home directories to be under the new ACSLS installation directory.

By default, the ACSLS installation directory is `/export/home` (referred to as `$BASEDIR`). The home directories for the ACSLS user IDs are:

- `acsss - $BASEDIR/ACSSS`
- `acssa - $BASEDIR/ACSSA`
- `informix - $BASEDIR/informix`

The following commands modify the above users' home directories. (You must be logged in as root.)

```
usermod -d $BASEDIR/ACSSS acsss
usermod -d $BASEDIR/ACSSA acssa
usermod -d $BASEDIR/informix informix
```

7. Reboot to verify that no database processes are running before you begin the install.

#### **Install using pkgadd:**

1. Log in as `root`.
2. Insert the ACSLS 7.1 CD-ROM.
3. In a terminal window or at the system prompt, enter

```
cd /cdrom/cdrom0
```

4. Install using `pkgadd`:

```
pkgadd -d .
```

**Note:** Make sure you enter a space and a period after `-d` `pkgadd` asks what package you want installed.

5. When prompted to select a package, select STKacsls and press `[[Return]]`.
6. Enter `y` or `n` at the prompt to install ACSLS in the default directory `/export/home/`.
7. Type `y` to at the prompt to install `setuid/setgid` files.
8. Select to continue at the super-user permission prompt.
9. Type `y` to at the prompt to install STKacsls.

User and group IDs are created (unless they already exist). Files being installed are displayed.

If the `acsss`, `acssa`, or `informix` user IDs are not defined with their home directories matching the ACSLS installation directory, the installation script displays a warning, for example:

```
***WARNING*** User acsss already exists, but its home
directory does not match the ACSLS installation
directory. Please change the acsss home directory to
/export/home/ACSSS after the installation.
```

If the `acsss`, `acssa`, or `informix` user IDs are created during installation, a default password is not created. You need to go into the `admintool` to create a password. These passwords must be maintained or set to never expire.

10. Eject the CD-ROM.

**eject cdrom**

### **Apply any desired maintenance to ACSLS.**

This ensures that any maintenance is applied before ACSLS is installed.

### **Install ACSLS software:**

1. Change directories:
 

```
cd /export/home/ACSSS/install
```
2. To initiate the installation shell script, enter
 

```
./install.sh
```

If shared memory settings have **not** been defined, you are prompted to allow the script to set shared memory and reboot the server:

This server is not set with shared memory required for ACSLS and the Database.

Set shared memory and reboot the server to take effect at kernel level? (y or n):

Respond y to the prompt.

The server reboots.

When the server comes back, log in as `root`, `cd` to `/export/home/ACSSS/` install (if you are not already in it), and restart `install.sh`.

3. Enter the database backup directory.

By default, this is `/export/backup`.

4. Respond y or n to the prompt for automatic startup on reboot.

**Option: If you have a SCSI or fibre-attached library continue with step 5**

5. Respond (y or n) to the prompt for installing a SCSI device driver for SCSI libraries.

Do you want to install the scsi device driver for SCSI libraries? (y or n):

---

YES The following message displays:

Enter the SCSI device(s) that correspond to each library. **Separate devices with a space (example: 4 5 6).** Remember that SCSI devices are numbers between 0 and 15.

**Notes:**

- On most Solaris machines, target-7 is not a valid device address because target-7 is reserved for the SCSI initiator on the host-bus adapter.
- If you are not sure what numbers to enter in this step follow the procedures in [“Troubleshooting a SCSI Connection” on page 516.](#)
- Continue with Step 6.

---

NO

---

**Note:**

Sun StorageTek libraries attached behind supported host-bus adapters (HBAs) can be auto-sensed by ACSLS using the capabilities included in supported HBA software. Supported HBAs currently include all contemporary Emulex, Qlogic, and Sun-branded HBAs. The ACSLS SCSI driver installation utility, *install\_scsi\_sol.sh* can configure multiple mchanger devices easily without the need for explicit user interaction. Libraries behind non-supported HBAs continue to function in the traditional manner where you declare the target and LUN address for each attached library. The installation utility then displays each library for which an mchanger instance has been created.

**Example**

```

Installing 64-bit mchanger
Probing for libraries...

One library found:
  STK L180 V-0310   Target 0 LUN 0

Are there additional libraries attached? (y or n): y

Enter the target:LUN pair corresponding to each library.
Separate target:LUN pairs with a space.
example: 4:0 5:0 5:1 5:2

==> 1:0 1:1

Use target 1 LUN 0
Use target 1 LUN 0

Is this correct? (y or n): y

Instances of 'mchanger' in /dev will be
built sequentially starting with mchanger 0.

Building an mchanger instance for each library...
Successfully built the following...
  /dev/mchanger0: STK L180 174-cells 4-drives
  /dev/mchanger1: STK L700 384-cells 8-drives
  /dev/mchanger2: STK SL500 65-cells 2-drives

```

Library driver installation is complete. You can now configure second-disk support.

**Configure second disk support**

Upon completion of the library driver installation, the following message displays:

```
install.sh 1501: You can enable/disable second disk
support later as root by doing the UNIX command "cd /
export/home/ACSSS/install; ./sd_mgr.sh".
```

You can enable second disk support after you complete the procedures in this chapter. See [Appendix A, "Second Disk Support."](#)

6. Set the ACSLS user passwords.

These passwords must be maintained or set to never expire. You need to set the following password to prevent a security exposure: `acsss`, `acssa`, `informix`, and `lib6`.

You must set the passwords the first time you login to these IDs. To set the passwords:

- a. Ensure you are logged in as user `root`.
- b. Enter a password for each user ID.  
Example: `passwd acsss`
- c. Confirm the password at the prompt.
- d. Login to the user IDs you use.
- e. Enter the password at the prompt.

7. If the `acsss`, `acssa`, or `informix` user IDs were not defined with their home directories matching the ACSLS installation directory, and the installation script displayed a warning, modify these user IDs so that their home directories are under the ACSLS base directory.

The following commands modify the above users' home directories. (You must be logged in as `root`.)

```
usermod -d $BASEDIR/ACSSS acsss
usermod -d $BASEDIR/ACSSA acssa
usermod -d $BASEDIR/informix informix
```

## ■ Step 6: Import the Database

In the following cases, you need to import the data from a previous ACSLS release. They are:

- If you are migrating from a previous release.
- If you are re-installing ACSLS.

Refer to [“Importing the Database” on page 209](#).

## ■ Step 7: Configure your Library Hardware

**CAUTION:** If you imported data from a previous ACSLS release, you must start ACSLS and ensure all LSMs are online before configuring your library hardware. Then shut down ACSLS. This initializes the LSM types and protects your imported database information.

**Note:** You do not need to run `acsss_config` if you are not changing your library hardware. Go to [“Step 8: Enter your License Key Information.”](#).

You must run `acsss_config` to configure you libraries if:

- this is a new installation,
- you are adding library hardware

Refer to [“Configuring or Updating your Library Hardware”](#) on page 130.

## ■ Step 8: Enter your License Key Information

You must enter a license key to activate ACSLS. To avoid unnecessary delays obtain your license key before you begin the ACSLS installation.

**Note:** You have a 75-day trial period during which you do not have to enter a license key. Please enter your license key before the trial period expires.

Refer to [Chapter 2, “ACSLs License Key”](#) for procedures.

## ■ Step 9: Reset any Custom Dynamic or Static Variables

If you are migrating to ACSLS 7.1 from a previous release and have customized your dynamic or static variables, you need to reset them.

## ■ Step 10: Configure the Second Disk

After installing and configuring ACSLS on the primary disk, configure the second disk as described in [Appendix A, “Second Disk Support.”](#)

## ■ Step 11: Audit new libraries

You need to audit your libraries:

- If this is a new installation.
- If you are adding new libraries to an existing configuration.

## ■ Step 12: Verify ACSLS Installation

Use the following procedure to mount or dismount a cartridge to verify ACSLS.

### Mount/dismount a cartridge to verify ACSLS:

1. Verify you are logged in as `acsss`.
2. If ACSLS is not running, start it by entering  
`rc.acsss`
3. Query the server from the `cmd_proc` by entering

```
query server
```

If messages are displayed indicating that the server is in recovery mode, wait for a message indicating that the server is running.

4. Verify that the following are online. You must have at least one of each online. If not, bring them online with the `vary` command.

```
query port all
```

```
query acs all
```

```
query lsm all
```

```
query drive all
```

5. Do you have at least one volume in an LSM?

---

YES Continue with the procedure.

---

NO Enter a volume into an LSM.

---

6. Mount a volume by entering:

```
mount vol_id drive_id
```

**Hint:** Use the `query drive` command to get the ID of an available drive and the `query volume` command to get the ID of a library volume. See [Chapter 14, "Command References."](#)

7. Did you see a message indicating a successful mount?

A successful mount message is:

```
Mount: vol_id mounted on drive_id
```

---

YES Procedure is complete.

---

**NO** If an error message appears, run this verification procedure again, ensuring that you specified a valid, available drive and a library volume. If the mount/dismount still fails, call StorageTek for assistance.

---

8. Dismount the volume by entering:

```
dismount vol_id drive_id force
```

where *vol\_id* is the volume and *drive\_id* is the drive you specified in Step 6.

## ■ ACSLS Setup for Client Communication

To set up ACSLS for client communication, you include the client host name and IP address in `/etc/hosts` or in the NIS lookup table.

## ■ Regressing to a Previous Version of ACSLS

If you need to regress to a previous version of ACSLS after you install ACSLS 7.1, you need to perform the following steps:

- Export the database and save it on a tape or file external to the server.
- Uninstall ACSLS 7.1 as described in the steps below:
- Install the previous version of ACSLS

For assistance in regressing to a previous version, contact StorageTek Software Support. For more information, see *Requesting Help from Software Support*.

## ■ Uninstalling ACSLS 7.1

**To uninstall ACSLS 7.1:**

1. Log in as `acsss`.
2. Shut down ACSLS.
 

```
kill.acsss
```
3. Shut down the Informix database:
 

```
db_command stop
```
4. Remove `pkgadd`:
  - Log in as `root`.
  - Enter `pkgrm STKacsls`

5. Perform a file cleanup for the primary disk:

- `cd /export/home`

If you installed ACSLS in another directory other than `/export/home`, `cd` to that directory.

- `rm -rf ACSSS informix`
- `cd /export/backup`

If you installed ACSLS backup in another directory other than `/export/backup`, `cd` to that directory.

- `rm -rf informix misc server_to_server`
- `cd /`
- `rm -rf INFORMIXTMP nsr`

6. Perform a file cleanup for the second disk:

- `cd /second_disk`

If you installed the second disk in another directory other than `/second_disk`, `cd` to that directory.

- `rm -rf data`
- `cd /second_disk/backup`

If you installed the second disk in another directory other than `/second_disk/backup`, `cd` to that directory.

- `rm -rf informix misc`

7. If you want to reinstall ACSLS in a different ACSLS installation directory, either delete the `acsss` and `informix` user id's or modify their home directories to be under the new ACSLS installation directory.

By default, the ACSLS installation directory is `/export/home` (referred to as `$BASEDIR`). The home directories for the ACSLS user IDs are:

- `acsss - $BASEDIR/ACSSS`
- `informix - $BASEDIR/informix`

The following commands modify the above users' home directories. (You must be logged in as root.)

```
usermod -d $BASEDIR/ACSSS acsss
usermod -d $BASEDIR/ACSSA acssa
usermod -d $BASEDIR/informix informix
```

8. Reboot.

# Installing ACSLS 7.1 on Solaris 10

## 4

---

This chapter describes procedures for installing ACSLS 7.1 on Sun Solaris 10 from a CD.

To apply any maintenance, refer to the installation instructions in the associated PUT or SPE or PTF document included with the CD, or the document associated with the maintenance tar file on the Customer Resource Center (CRC) web page.

This chapter takes you through and discusses the following steps for installing ACSLS on Solaris 10. In these steps:

1. Have your license key information and ACSLS 7.1 CD available.
2. Export the database if you are upgrading from a previous version of ACSLS.
3. Prepare for Solaris 10 installation.
4. Install the Solaris 10 operation system.
5. Prepare for ACSLS installation.
6. Install ACSLS.
7. Import the database from any previous version.  
**Note:** If you import the database, you can skip the next step.
8. Run `acsss_config` to configure your library hardware (see [“Configuring or Updating your Library Hardware” on page 130](#)).
9. Enter your license key information (see [Chapter 2, “ACSLs License Key”](#)). A 75-day trial license is automatically installed when ACSLS is initially installed.
10. Reset any custom dynamic or static variables. Performing an import and selecting option 4 merges any previous customized dynamic variables.
11. Audit new libraries.
12. Verify the ACSLS installation.

In addition, this chapter discusses:

- ACSLS Setup for Client Communication

- Regressing to a previous version of ACSLS
- Uninstalling ACSLS 7.1

## ■ Step 1: Export the Database

Perform this step if you are upgrading from a previous version of ACSLS (see [“Exporting the Database” on page 206](#)).

## ■ Step 2: Prepare for Solaris Installation

Before installing Solaris, complete the following steps:

1. Have your license key information and ACSLS 7.1 CD available.
2. Satisfy all requirements specified in *Product Information* (see [“ACSL S Information on the StorageTek CRC” on page xl](#)) prior to installation of Solaris or ACSLS.

ACSL S 7.1 requires a minimum of 4GB on the primary disk system.

3. Install the library hardware according to the Sun StorageTek installation instructions.

**Note:** If you are installing a SCSI library, you should connect the server to the library before installing ACSLS. The `install.sh` script installs device drivers and kernel patches. If the SCSI cable is not connected, the device driver installation fails. However, if library hardware is not available at the time of software installation, the driver can be installed separately at a later time.

4. Install the ACSLS server hardware according to the hardware manufacturer's instructions.
5. Determine and record information about network communication settings.

## Pre-Installation Checklist

The following checklist provides the information that you need when installing the Solaris 10 operating system. During installation, you only need to answer the prompts that pertain to your operating system.

**CAUTION:** Consult your system administrator for assistance in obtaining this information. It is critical that this information is correct in order for ACSLS to run properly.

**Table 4. Description of Installation Prompts**

Prompt/Task	Description	Your Values
<input type="checkbox"/> Network connection	Is the system connected to a network	
<input type="checkbox"/> DHCP  If you don't use DHCP you are prompted to enter the: <input type="checkbox"/> IP Address <input type="checkbox"/> Subnet Mask <input type="checkbox"/> Whether to enable IPv6.	Can the system use Dynamic Host Configuration Protocol (DHCP) to configure the network interface?	
<input type="checkbox"/> Host Name	Host name that you choose for the system.	
<input type="checkbox"/> Internet Protocol Address	Enter the IP address if you do not use DHCP to configure the network interface.	
<input type="checkbox"/> Netmask	Enter the subnet mask pattern if you do not use DHCP to configure the network interface.	
<input type="checkbox"/> IPv6	Do you want to enable IPv6 on this machine?	
<input type="checkbox"/> Kerberos	Select whether to configure Kerberos security on this machine. If you do this, you must gather the following:  <input type="checkbox"/> Default Realm <input type="checkbox"/> Administration server <input type="checkbox"/> First KDC <input type="checkbox"/> Additional KDCs (option)	

**Table 4. Description of Installation Prompts**

Prompt/Task	Description	Your Values
<p><input type="checkbox"/> Name Service</p> <p>If you provide a name service, you must also supply the following:</p> <p style="padding-left: 40px;"><input type="checkbox"/> Domain Name</p> <p style="padding-left: 40px;"><input type="checkbox"/> NIS+ or NIS</p> <p style="padding-left: 40px;"><input type="checkbox"/> DNS</p> <p style="padding-left: 40px;"><input type="checkbox"/> LDAP</p>	<p>Name of service the system should use.</p> <p>Do not select NIS+ or NIS services unless your system administrator can configure these services correctly.</p> <p>You will be asked to specify a name server or allow the system to select one.</p> <p>If you want to specify a service, you must supply the server's host name and the server's IP address.</p> <p>Do not select the DNS service unless your system administrator can configure these services correctly.</p> <p>Enter the IP addresses for the DNS server. You must enter at least one IP address, but you can enter up to three addresses.</p> <p>Enter your LDAP information (profile name and profile server).</p> <p>If you specify a proxy credential level in your LDAP enter the Proxy-Bind Distinguished Name and Proxy-Bind Password.</p>	

**Table 4. Description of Installation Prompts**

Prompt/Task	Description	Your Values
<input type="checkbox"/> Default Route  Select one of the following: <input type="checkbox"/> Detect one upon reboot <input type="checkbox"/> Specify one <input type="checkbox"/> None	Do you want to specify a default route IP address or allow Solaris installation program find one?  The default route provides a bridge that forwards traffic between two physical networks. The IP address is a unique number that identifies each host on a network.  <input type="checkbox"/> Detect one upon reboot (Solaris installation program will detect an IP address) <input type="checkbox"/> Specify one (you specify the IP address) <input type="checkbox"/> None (if you do not have a router or do not want the system to detect an IP address).	
<input type="checkbox"/> Time Zone	You will be asked to specify your default time zone. Selections are: Geographic Region; Offset from GMT; and Time Zone File.	
<input type="checkbox"/> Root Password	This is the root password for this system.	
<input type="checkbox"/> Type of Installation  Select one of the following: <input type="checkbox"/> Standard <input type="checkbox"/> Flash	Standard installs from the standard Solaris distribution. This allows you to choose between an initial or upgrade install.  Flash installs your system from one or more Flash archives.  <b>Select standard.</b>	
<input type="checkbox"/> Locale	This is the geographic regions where you want to install support.	
<input type="checkbox"/> Disk Selection	This selects the disk on which you want to install Solaris software.	

**Table 4. Description of Installation Prompts**

Prompt/Task	Description	Your Values
<input type="checkbox"/> Proxy Server Configuration	Do you have a direct connection to the Internet or do you need a proxy server to access to the Internet. If you use a proxy server, you must provide the host and port.	
<input type="checkbox"/> Software Group	What software groups do you want to install?  <input type="checkbox"/> Entire <input type="checkbox"/> Developer <input type="checkbox"/> Documentation <input type="checkbox"/> Java	
<input type="checkbox"/> Preserve Data	Allows you to preserve any data that exists on the disks where you are installing the Solaris software.	
<input type="checkbox"/> Layout File Systems Select one of the following:  <input type="checkbox"/> Auto Layout <input type="checkbox"/> Manual	Auto Layout allows the installation program to automatically lay out file systems on your disks. Manual allows you to customize your disk partitions.  <b>Select Manual.</b>	

## ■ Step 3: Install the Solaris 10 Operating System

The following instructions apply only if you have purchased the ACSLS turnkey hardware platform. In this case, you receive the official Solaris distribution on DVD media.

For any other installations, such as downloading Solaris from the Sun website, you must refer to their Solaris installation manuals.

1. Insert the Solaris 10 Operating System into the DVD drive.
  - a. su to root.
  - b. reboot with DVD  
Press <Enter>.

**Note:** If you have configured the SCSI/Ethernet card on your server and you are not using the second Ethernet connection, then disregard the message, “home1: link down”.

The `Solaris Install Console` will appear first. During the installation process, this console provides messages indicating what the system is doing.

2. Select the language to use during the installation and click F2.

A Welcome screen appears with configuration information.

3. Click F2 to begin the installation.

A Welcome screen appears with configuration information.

4. Click F2.

You are then guided through several configuration screens. Refer to [Table 4 on page 51](#) for a description of these screen prompts. You can also use this table to enter your values beforehand.

5. Press F2 on the `Confirm Information` window.

6. Continue answering the configuration prompts.

**Note:**

- New install or new system:

On a new system, or if you are installing Solaris 10 for the first time, select *initial* installation.

- Layout the Filesystem:

The Layout of the Filesystem screen allows you to customize your disk partitions. Select *Manual*.

7. Set the size of your Disk Partitions by clicking F4.

How you set up your disk partitions can have a significant impact on system performance. This is because disk I/O is used for file system operations and also by the virtual memory subsystem for paging and swapping.

**Note:**

- Assign the mount point for each of the partitions in the left column.
- Assign the size for each partition in the right column except for `/export/backup`.

You may be able to improve disk I/O performance by following the configuration guidelines described in the figures below.

It is recommend that you have a minimum disk size of 20 GB. The following figure is a guideline for partitioning a 20 GB disk.

**Figure 7. Primary Disk Partitions Guidelines**

Disk: c0t0d0		Disk Size: 20 GB
		Minimum Requirements
0	/	237 MB
1	swap	513 MB
2	overlay This is your total disk size.	
3	/var	2600 MB
4	/export/home	2000 MB
5	/opt	4542 MB
6	/usr	4524 MB
7	/export/backup	The remaining disk memory should be assigned to this partition

8. Click F2 after entering the disk partitions.  
The Lay Out File System screen refreshes with the values you entered.
9. If the information is correct, click F2.  
A Mount Remote File System screen appears.
10. Click F2.  
The Profile screen appears displaying the information you entered.
11. Click F2.  
The system begins the installation.  
Upon successful installation, reboot.
12. Continue with the installation by inserting the appropriate disks.  
You are prompted to insert additional disks based on the Software Groups you selected.  
When the installation is complete the Welcome screen appears.
13. Enter your user name and click F2.
14. Enter the password and click F2.

15. Select Desktop and click F2.

## ■ Step 4: Prepare for ACSLS Installation

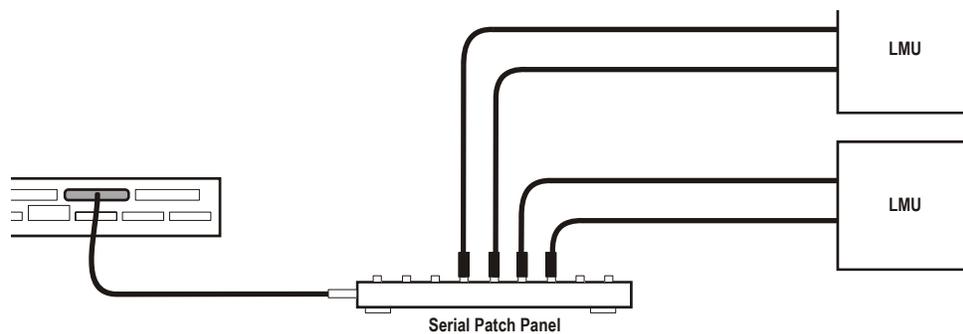
Before you install ACSLS, complete the following steps:

1. Verify that the server system hardware is properly configured, connected, powered on, and ready.
2. Connect the LMU to a valid serial port.

If more than one port is available on your server, configure two ports to the LMU. An alternate LMU connection provides higher throughput and greater robustness.

3. Multiple serial port connects are necessary if you are configuring the server to a dual-LMU installation. You can provide redundant connections to each LMU using a multi-port adapter as suggested in [Figure 8](#).

**Figure 8. Dual LMU Connection Configuration**



S403\_003

4. If your installation includes a SL8500 library or a 9300 library and TCP/IP LMU(s), connect the TCP/IP LMU(s) to the network used for LMU communication.
5. For SCSI-connected libraries, you should use a differential connection where possible. If a single-ended SCSI controller is used, you should limit the cable distance to three meters between the server and the library. With low-voltage differential (LVD), the cable should be no more than 10 meters. High-voltage differential (HVD) SCSI cables can extend up to 20 meters.
6. Make sure that each attached LMU and LSM is fully configured, powered on, and ready.

**Note:** The configuration script, `acsss_config`, will fail unless all LMUs and LSMs are fully configured, powered on, and ready.

7. If you have any communication problems refer to the troubleshooting chapter in the appendix.

8. If you intend to use removable media for database backup, have a blank cartridge available for your backup tape device to complete the configuration process.
9. ACSLS requires specific user IDs. If these user IDs are not defined before ACSLS installation, ACSLS creates them. When ACSLS installation creates the user IDs, the system assigns the user ID numbers and group ID numbers. To assign specific user ID and group ID numbers, you must define the following groups and users before installing ACSLS:

<b>User</b>	<b>Group</b>
acsss	staff
acssa	staff
lib6	staff
informix	informix

When these user IDs are defined before ACSLS installation, they can either be defined locally (on the ACSLS server) or via remote authentication methods (e.g., NIS or Kerberos). These user IDs must be defined with the following properties:

- The shell for these user IDs must be sh on Solaris. The exception is acssa (ksh).
- The home directories for the acsss, acssa, and informix user IDs must be under the ACSLS installation directory. By default, the ACSLS installation directory is /export/home (referred to as \$BASEDIR). The home directories for the ACSLS user IDs are:

acsss	\$BASEDIR/ACSSS
acssa	\$BASEDIR/ACSSA
informix	\$BASEDIR/informix
lib6	\$BASEDIR/ACSSS

If the home directories for the ACSLS user IDs do not match their required locations, please either modify the home directories for these users or delete the user IDs so they are added correctly during the ACSLS installation process.

The following command creates the `acsss` user on Solaris. (You must be logged in as `root`.)

```
useradd -d $BASEDIR/ACSSS -g staff -s /bin/sh -c "ACSLs Control Login" acsss
```

The account information is:

```
acsss -d $BASEDIR/ACSSS-g staff -c "ACSLs Control Login"-s /bin/sh
acssa -d $BASEDIR/ACSSA-g staff -c "ACSLs SA Login"-s /bin/ksh
informix -d $BASEDIR/informix/IDS7.3-g informix-c "ACSLs Informix Database Owner"-s /bin/sh
lib6 -d $BASEDIR/ACSSS-g staff -s /bin/sh
```

The following commands modify the `acsss`, `acssa`, and `informix` users' home directories. (You must be logged in as `root`.)

```
usermod -d $BASEDIR/ACSSS acsss
usermod -d $BASEDIR/ACSSA acssa
usermod -d $BASEDIR/informix informix
```

## ■ Step 5: Install ACSLS

### Remove any previous version

1. Have your license key available.
2. Is this a new installation?

---

YES    Go to ["Install pkgadd" on page 62](#).

---

NO     Make sure you exported the database by using the `db_export.sh` utility command.

---

3. Shut down ACSLS:

You need to be logged in as `acsss` to do this.

- a. Enter the following:

```
idle (from a cmd_proc)
```

```
kill.acsss (from a command prompt)
```

- b. Enter one of the following, depending on the ACSLS release from which you are upgrading:

- If you are upgrading from ACSLS 5.3 or 5.3.2:

```
db_command stop
```

- If you are upgrading from ACSLS 5.4:

```
db_command.sh stop
```

- If you are upgrading from ACSLS 6.0 and later:

### **db\_command stop**

4. Login as `root` and remove ACSLS, backup and other files:

For ACSLS 5.3 or 5.3.2, enter:

- `cd /export/home`
- `rm -rf ACSSS oracle`
- `cd /export/backup`
- `rm -rf oracle misc`

For ACSLS 5.4, enter:

- `cd /export/home`
- `rm -rf ACSSS sybase`
- `cd /export/backup`
- `rm -rf sybase misc`

For ACSLS 6.0 or 6.0.1, enter:

- `cd /export/home`
- `rm -rf ACSSS informix`
- `cd /export/backup`
- `rm -rf informix misc`

If the `server_to_server` directory exists, enter:

```
rm -rf informix misc server_to_server
```

- `cd /`
- `rm -rf INFORMIXTMP nsr`

For ACSLS 6.1.0, 6.1.1, or 7.0 enter:

- `cd /export/home`
- `pkgrm STKacsls`
- `rm -rf ACSSS ACSSA informix`
- `cd /export/backup`
- `rm -rf informix misc server_to_server`
- `cd /`
- `rm -rf INFORMIXTMP`
- `rm -rf /nsr`

- `cd /var/tmp`
- `rm -rf acsls`

5. Remove files under second disk (if installed)

- `cd /second_disk`

If you installed the second disk in another directory other than `/second_disk`, cd to that directory.

- `rm -rf data`
- `cd /second_disk/backup`

If you installed the second disk in another directory other than `/second_disk/backup`, cd to that directory.

For ACSLS 5.3 or 5.3.2, enter:

- **`rm -rf oracle misc`**

For ACSLS 5.4, enter:

- `rm -rf sybase misc`

For ACSLS 6.0 and later enter:

- `rm -rf informix misc`

6. If you want to reinstall ACSLS in a different ACSLS installation directory, either delete the `acsss`, `acssa`, and `informix` user IDs or modify their home directories to be under the new ACSLS installation directory.

By default, the ACSLS installation directory is `/export/home` (referred to as `$BASEDIR`). The home directories for the ACSLS user IDs are:

- `acsss - $BASEDIR/ACSSS`
- `acssa - $BASEDIR/ACSSA`
- `informix - $BASEDIR/informix`

The following commands modify the above users' home directories. (You must be logged in as root.)

```
usermod -d $BASEDIR/ACSSS acsss
usermod -d $BASEDIR/ACSSA acssa
usermod -d $BASEDIR/informix informix
```

7. Reboot to verify that no database processes are running before you begin the install.

## Install pkgadd

1. Log in as `root`.
2. Insert the ACSLS 7.1 CD.
3. In a terminal window or at the system prompt, enter  
**cd /cdrom/cdrom0**
4. Install using `pkgadd`:  
`pkgadd -d .`  
**Note:** Make sure you enter a space and a period after `-d`  
`pkgadd` asks what package you want installed.
5. When prompted to select a package, select `STKacsls` and press `[[Return]]`.
6. Enter `y` or `n` at the prompt to install ACSLS in the default directory `/export/home/`.  
Use the default directory `/export/home` or enter the directory where you want ACSLS installed.
7. Type `y` to at the prompt to install `setuid/setgid` files.
8. Select to continue at the super-user permission prompt.
9. Type `y` to at the prompt to install `STKacsls`.  
User and group IDs are created (unless they already exist). Files being installed are displayed.  
If the `acsss`, `acssa`, or `informix` user IDs are not defined with their home directories matching the ACSLS installation directory, the installation script displays a warning, for example:  

```
***WARNING*** User acsss already exists, but its home directory does not match the ACSLS installation directory. Please change the acsss home directory to /export/home/ACSSS after the installation.
```

  
If the `acsss`, `acssa`, or `informix` user IDs are created during installation, a default password is not created. You need to go into the `admintool` to create a password. These passwords must be maintained or set to never expire.
10. Eject the CD.  
**eject cdrom**

**Apply any desired maintenance to ACSLS**

This ensures that any maintenance is applied before ACSLS is installed.

**Install ACSLS software**

1. Change directories:

```
cd /export/home/ACSSS/install
```

2. To initiate the installation shell script, enter

```
./install.sh
```

**Note:** Should you get the message below, type **y** since Solaris 10 is supported.

```
Warning: Solaris 5.10 may not be supported - you should check
the CRC or call Support to find out before proceeding.
Continue with install? (y or n):
```

If shared memory settings have **not** been defined, you are prompted to allow the script to set shared memory and reboot the server:

```
This server is not set with shared memory required for
ACSLs and the Database.
```

```
Set shared memory and reboot the server to take effect
at kernel level? (y or n):
```

Respond **y** to the prompt.

The server reboots.

When the server comes back, log in as `root`, `cd` to `/export/home/ACSSS/install` (if you are not already in it), and restart `install.sh`.

3. Enter the database backup directory.

By default, this is `/export/backup`.

4. Respond **y** or **n** to the prompt for automatic startup on reboot.

**Option: If you have a SCSI or fibre-attached library continue with step 5.**

5. Respond (**y** or **n**) to the prompt for installing a SCSI device driver for SCSI libraries.

Do you want to install the scsi device driver for SCSI libraries? (y or n):

---

YES The following message displays:

Enter the SCSI device(s) that correspond to each library. **Separate devices with a space (example: 4 5 6).** Remember that SCSI devices are numbers between 0 and 15.

**Notes:**

- On most Solaris machines, target-7 is not a valid device address because target-7 is reserved for the SCSI initiator on the host-bus adapter.
- If you are not sure what numbers to enter in this step follow the procedures in [“Troubleshooting a SCSI Connection” on page 516.](#)
- Continue with Step 6.

---

NO

---

**Note:**

Sun StorageTek libraries attached behind supported host-bus adapters (HBAs) can be auto-sensed by ACSLS using the capabilities included in supported HBA software. Supported HBAs currently include all contemporary Emulex, Qlogi, and Sun-branded HBAs. The ACSLS SCSI driver installation utility, *install\_scsi\_sol.sh* can configure multiple mchanger devices easily without the need for explicit user interaction. Libraries behind non-supported HBAs continue to function in the traditional manner where you declare the target and LUN address for each attached library. The installation utility then displays each library for which an mchanger instance has been created.

**Example**

```
Installing 64-bit mchanger
Probing for libraries...
```

```
One library found:
  STK L180 V-0310   Target 0 LUN 0
```

```
Are there additional libraries attached? (y or n): y
```

```
Enter the target:LUN pair corresponding to each library.
Separate target:LUN pairs with a space.
example: 4:0 5:0 5:1 5:2
```

```
==> 1:0 1:1
```

```

Use target 1 LUN 0
Use target 1 LUN 0

Is this correct? (y or n): y

Instances of 'mchanger' in /dev will be
built sequentially starting with mchanger 0.

Building an mchanger instance for each library...
Successfully built the following...
    /dev/mchanger0: STK L180 174-cells 4-drives
    /dev/mchanger1: STK L700 384-cells 8-drives
    /dev/mchanger2: STK SL500 65-cells 2-drives

```

Library driver installation is complete. You can now configure second-disk support.

### Configure second disk support

Upon completion of the library driver installation, the following message displays:

```
install.sh 1501: You can enable/disable second disk support
later as root by doing the UNIX command "cd /export/home/
ACSSS/install; ./sd_mgr.sh".
```

You can enable second disk support after you complete the procedures in this chapter. See [Appendix A, "Second Disk Support."](#)

#### 6. Set the ACSLS user passwords.

These passwords must be maintained or set to never expire. You need to set the following password to prevent a security exposure: `acsss`, `acssa`, `informix`, and `lib6`.

You must set the passwords the first time you login to these IDs. To set the passwords:

- a. Login to each of the above user IDs.
- b. Enter the password at the prompt.

#### 7. If the `acsss`, `acssa`, or `informix` user IDs were not defined with their home directories matching the ACSLS installation directory, and the installation script displayed a warning, modify these user IDs so that their home directories are under the ACSLS base directory.

The following commands modify the above users' home directories. (You must be logged in as `root`.)

```

usermod -d $BASEDIR/ACSSS acsss
usermod -d $BASEDIR/ACSSA acssa
usermod -d $BASEDIR/informix informix

```

## ■ Step 6: Import the Database

In the following cases, you need to import the data from a previous ACSLS release. They are:

- If you are migrating from a previous release.
- If you are re-installing ACSLS.

Refer to [“Importing the Database” on page 209](#).

## ■ Step 7: Configure your Library Hardware

**CAUTION:** If you imported data from a previous ACSLS release, you must start ACSLS and ensure all LSMs are online before configuring your library hardware. Then shut down ACSLS. This initializes the LSM types and protects your imported database information.

**Note:** You do not need to run `acsst_config` if you are not changing your library hardware. Go to [“Step 8: Enter your License Key Information.”](#).

You must run `acsst_config` to configure your libraries if:

- this is a new installation,
- you are adding library hardware

Refer to [“Configuring or Updating your Library Hardware” on page 130](#).

## ■ Step 8: Enter your License Key Information

You must enter a license key to activate ACSLS. To avoid unnecessary delays obtain your license key before you begin the ACSLS installation.

**Note:** You have a 75-day trial period during which you do not have to enter a license key. Please enter your license key before the trial period expires.

Refer to [Chapter 2, “ACSLs License Key”](#) for procedures.

## ■ Step 9: Reset any Custom Dynamic or Static Variables

If you are migrating to ACSLS 7.1 from a previous release and have customized your dynamic or static variables, you need to reset them.

## ■ Step 10: Configure the Second Disk

After installing and configuring ACSLS on the primary disk, configure the second disk as described in [Appendix A, “Second Disk Support.”](#)

## ■ Step 11: Audit new libraries

You need to audit your libraries:

- If this is a new installation.
- If you are adding new libraries to an existing configuration.

## ■ Step 12: Verify ACSLS Installation

Use the following procedure to mount or dismount a cartridge to verify ACSLS.

### Mount/dismount a cartridge to verify ACSLS:

1. Verify you are logged in as `acsss`.
2. If ACSLS is not running, start it by entering
 

```
rc.acsss
```
3. Query the server from the `cmd_proc` by entering
 

```
query server
```

If messages are displayed indicating that the server is in recovery mode, wait for a message indicating that the server is running.
4. Verify that the following are online. You must have at least one of each online. If not, bring them online with the `vary` command.
 

```
query port all
query acs all
query lsm all
query drive all
```
5. Do you have at least one cartridge in an LSM?
 

YES	Continue with the procedure.
NO	Enter a cartridge into an LSM.
6. Mount a volume by entering:
 

```
mount vol_id drive_id
```

**Hint:** Use the `query drive` command to get the ID of an available drive and the `query volume` command to get the ID of a library cartridge. See [Chapter 14, “Command References.”](#)

7. Did you see a message indicating a successful mount?

A successful mount message is:

```
Mount: vol_id mounted on drive_id
```

---

YES	Procedure is complete.
NO	If an error message appears, run this verification procedure again, ensuring that you specified a valid, available drive and a library cartridge. If the mount/dismount still fails, call StorageTek for assistance.

---

8. Dismount the cartridge by entering:

```
dismount vol_id drive_id force
```

where *vol\_id* is the volume and *drive\_id* is the drive you specified in [Step 6](#).

## ■ ACSLS Setup for Client Communication

To set up ACSLS for client communication, you include the client host name and IP address in `/etc/hosts` or in the NIS lookup table.

## ■ Regressing to a Previous Version of ACSLS

If you need to regress to a previous version of ACSLS after you install ACSLS 7.1, you need to perform the following steps:

- Export the database and save it on a tape or file external to the server.
- Uninstall ACSLS 7.1 as described in the steps below:
- Install the previous version of ACSLS

For assistance in regressing to a previous version, contact StorageTek Software Support. For more information, see *Requesting Help from Software Support*.

## ■ Uninstalling ACSLS 7.1

### To uninstall ACSLS 7.1:

1. Log in as `acsss`.
2. Shut down ACSLS.  
`kill.acsss`
3. Shut down the Informix database:  
`db_command stop`
4. Remove `pkgadd`:
  - Log in as `root`.
  - Enter `pkgrm STKacsls`
5. Perform a file cleanup for the primary disk:
  - `cd /export/home`

If you installed ACSLS in another directory other than `/export/home`, `cd` to that directory.

- `rm -rf ACSSS informix`
- `cd /export/backup`

If you installed ACSLS backup in another directory other than `/export/backup`, `cd` to that directory.

- `rm -rf informix misc server_to_server`
- `cd /`
- `rm -rf INFORMIXTMP nsr`

6. Perform a file cleanup for the second disk:

- `cd /second_disk`

If you installed the second disk in another directory other than `/second_disk`, `cd` to that directory.

- `rm -rf data`
- `cd /second_disk/backup`

If you installed the second disk in another directory other than `/second_disk/backup`, `cd` to that directory.

- `rm -rf informix misc`

7. If you want to reinstall ACSLS in a different ACSLS installation directory, either delete the acsss and informix user id's or modify their home directories to be under the new ACSLS installation directory.

By default, the ACSLS installation directory is /export/home (referred to as \$BASEDIR). The home directories for the ACSLS user IDs are:

- acsss - \$BASEDIR/ACSSS
- informix - \$BASEDIR/informix

The following commands modify the above users' home directories. (You must be logged in as root.)

```
usermod -d $BASEDIR/ACSSS acsss
usermod -d $BASEDIR/ACSSA acssa
usermod -d $BASEDIR/informix informix
```

8. Reboot.

## ■ Running ACSLS within Solaris Zones

Management of StorageTek SCSI libraries within Solaris Zones is supported by ACSLS when you install the STKchanger driver package in the global zone.

Zone technology is ideal for environments that consolidate a number of applications on a single server, thus allowing you to create multiple virtual environments on a single system so applications can safely run without endangering each other. Zone partitioning allows you to isolate these software applications and services by setting up boundaries between zones so that you can dynamically control application and resource priorities.

There are two types of zones:

- Global zone

A global zone is the default zone for the system and is used for system-wide administrative control. It is from this zone, that non-global zones can be configured.

The STKchanger resides in the global zone.

- Non-global zone (container)

Non-global zones partitioning allows you to create multiple private execution environments and to dynamically control applications and resources. This isolation prevents processes that are running in one zone from monitoring or affecting processes that are running in other zones. Even a process running with superuser credentials cannot view or affect activity in other zones.

The ACSLS server resides in the non-global zone(s).

**Note:** One instance of ACSLS is supported for each non-global zone. ACSLS **must be** the only application running in that particular container.

Refer to the *Sun System Administration Guide: Solaris Containers-Resource Management and Solaris Zones* for information on configuring and managing Solaris zones (containers).

Refer to [“Running ACSLS within Solaris Zones” on page 70](#) for the installation procedures.

To support a container environment, the STK SCSI Media Changer drivers and utilities now reside in a standalone package. This is due to the fact that devices for any container environment are installed and controlled at the global zone level. Prior to Solaris 10, the mchanger device drivers were included within the ACSLS application only. Since Solaris 10 includes the ability to create several different server instances on one hardware platform, and in order to make mchanger devices available on non-global containers, the mchanger devices must first be created within the global zone and then made available to the container. Since only the mchanger package is required within the global zone, a new package, STKchanger, has been created. This allows mchanger to be installed in the global zone without having to install ACSLS. To do this you need to:

- Install STKchanger in the global zone.
- Make it available to the proper container.
- Install ACSLS within the non-global container.
- Use the mchanger device that is available within the non-global zone.

## Installing the STKchanger

STKchanger is installed using the `pkgadd` commands and requires that it be installed as the user root.

1. Log in as `root` to the Global Zone.
2. Insert the ACSLS PUT0602S CD
3. In a terminal window or at the system prompt, enter

```
cd /cdrom/cdrom0
```

4. Install using `pkgadd`:

```
pkgadd -d .
```

**Note:** A space and a period must be entered after the command.

5. When prompted to select a package, select STKchanger and press `[[Return]]`.

The STKchanger package is then installed in `/opt/STKchanger`.

6. Close and reopen the window.
7. Eject the CD.

## Connecting to a library

1. Log in as `root` to the Global Zone.
2. Change to the STKchanger installation directory

```
cd /opt/STKchanger/bin
```

3. Run `install.sh`

```
./install.sh
```

### Example:

```
Installing 64-bit mchanger
Probing for libraries...
One library found:
  STK L180 V-0310 Target 0 LUN 0
Are there additional libraries attached? (y or n): y
Enter the target:LUN pair corresponding to each library.
Separate target:LUN pairs with a space.
example: 4:0 5:0 5:1 5:2

==> 1:0 1:1
Use target 1 LUN 0
Use target 1 LUN 1
Is this correct? (y or n): y
Instances of 'mchanger' in /dev will be
built sequentially starting with mchanger0.
Building an mchanger instance for each library...
Successfully built the following...

/dev/mchanger0: STK L180 174-cells 4-drives
/dev/mchanger1: STK L700 384-cells 8-drives
/dev/mchanger2: STK SL500 65-cells 2-drives
```

You now have three available mchanger devices. You can assign them to specific non-global zones (containers) using the `zonecfg` command.

Refer to the *Sun System Administration Guide: Solaris Containers-Resource Management and Solaris Zones* for information on the `zonecfg` command.

The following utilities included within the STKchanger package are:

- `showDevs.sh`
- `probeFibre.sh`

Refer to [Chapter 13, Utility Reference](#) for more information about these utilities.



# Installing ACSLS 7.1.1 on Solaris 10 x86

## 5

---

This chapter describes procedures for installing ACSLS 7.1.1 on the Sun Solaris 10 x86 system.

- ACSLS 7.1.1 supports only the Solaris 10 x86 platform.
- This release does not support Soaris Zones (containers).

This chapter takes you through and discusses the following steps for installing ACSLS on Solaris 10. In these steps:

1. Have your license key information and ACSLS 7.1.1 CD available.
2. Export the database if you are upgrading from a previous version of ACSLS.
3. Prepare for Solaris 10 installation.
4. Install the Solaris 10 operation system.
5. Prepare for ACSLS installation.
6. Install ACSLS.
7. Import the database from any previous version.  
**Note:** If you import the database, you can skip the next step.
8. Run `acsss_config` to configure your library hardware (see [“Configuring or Updating your Library Hardware” on page 130](#)).
9. Enter your license key information (see [Chapter 2, “ACSLs License Key”](#)). A 75-day trial license is automatically installed when ACSLS is initially installed.
10. Reset any custom dynamic or static variables. Performing an import and selecting option 4 merges any previous customized dynamic variables.
11. Audit new libraries.
12. Verify the ACSLS installation.

In addition, this chapter discusses:

- ACSLS Setup for Client Communication
- Uninstalling ACSLS 7.1.1

## ■ Step 1: Export the Database

Perform this step if you are upgrading from a previous version of ACSLS (see [“Exporting the Database” on page 206](#)).

## ■ Step 2: Preparing for Solaris Installation

Before installing Solaris, complete the following steps:

1. Have your license key information and ACSLS 7.1.1 CD available.
2. Satisfy all requirements specified in *Product Information* (see [“ACSLS Information on the StorageTek CRC” on page xl](#)) prior to installation of Solaris or ACSLS.

ACSLS 7.1.1 requires a minimum of 4GB on the primary disk system.

3. Install the library hardware according to the Sun StorageTek installation instructions.

**Note:** If you are installing a SCSI library, you should connect the server to the library before installing ACSLS. The `install.sh` script installs device drivers and kernel patches. If the SCSI cable is not connected, the device driver installation fails. However, if library hardware is not available at the time of software installation, the driver can be installed separately at a later time.

4. Install the ACSLS server hardware according to the hardware manufacturer's instructions.
5. Determine and record information about network communication settings.

## Pre-Installation Checklist

The following checklist provides the information that you need when installing the Solaris 10 operating system. During installation, you only need to answer the prompts that pertain to your operating system.

**CAUTION:** Consult your system administrator for assistance in obtaining this information. It is critical that this information is correct in order for ACSLS to run properly.

**Table 5. Description of Installation Prompts**

Prompt/Task	Description	Your Values
<input type="checkbox"/> Network connection	Is the system connected to a network	
<input type="checkbox"/> DHCP  If you don't use DHCP you are prompted to enter the: <input type="checkbox"/> IP Address <input type="checkbox"/> Subnet Mask <input type="checkbox"/> Whether to enable IPv6.	<b>Do not use</b> Dynamic Host Configuration Protocol (DHCP) for the ACSLS server.	
<input type="checkbox"/> Host Name	Host name that you choose for the system.	
<input type="checkbox"/> Internet Protocol Address	Enter the IP address if you do not use DHCP to configure the network interface.	
<input type="checkbox"/> Netmask	Enter the net mask if you do not use DHCP to configure the network interface.	
<input type="checkbox"/> IPv6	Do you want to enable IPv6 on this machine?	

**Table 5. Description of Installation Prompts**

Prompt/Task	Description	Your Values
<input type="checkbox"/> Kerberos	<p>Select whether to configure Kerberos security on this machine. If you do this, you must gather the following:</p> <ul style="list-style-type: none"> <li><input type="checkbox"/> Default Realm</li> <li><input type="checkbox"/> Administration server</li> <li><input type="checkbox"/> First KDC</li> <li><input type="checkbox"/> Additional KDCs (option)</li> </ul> <p>Kerberos should <b>not</b> be used as the network authentication protocol.</p>	

**Table 5. Description of Installation Prompts**

Prompt/Task	Description	Your Values
<p><input type="checkbox"/> Name service</p> <p>If you provide a name service, you must also supply the following:</p> <p style="padding-left: 40px;"><input type="checkbox"/> Domain Name</p> <p style="padding-left: 40px;"><input type="checkbox"/> NIS+ or NIS</p> <p style="padding-left: 40px;"><input type="checkbox"/> DNS</p> <p style="padding-left: 40px;"><input type="checkbox"/> LDAP</p> <p style="padding-left: 40px;"><input type="checkbox"/> None</p>	<p>Name of service the system should use.</p> <p>Do not select NIS+ or NIS services unless your system administrator can configure these services correctly.</p> <p>You will be asked to specify a name server or allow the system to select one.</p> <p>If you want to specify a service, you must supply the server's host name and the server's IP address.</p> <p>Do not select the DNS service unless your system administrator can configure these services correctly.</p> <p>Enter the IP addresses for the DNS server. You must enter at least one IP address, but you can enter up to three addresses.</p> <p>Enter your LDAP information (profile name and profile server).</p> <p>If you specify a proxy credential level in your LDAP enter the Proxy-Bind Distinguished Name and Proxy-Bind Password.</p> <p>Either a service name is not used or the name of the service used does not appear in this list.</p>	

**Table 5. Description of Installation Prompts**

Prompt/Task	Description	Your Values
<p><input type="checkbox"/> Default Route</p> <p>Select one of the following:</p> <p><input type="checkbox"/> Detect one upon reboot</p> <p><input type="checkbox"/> Specify one</p> <p><input type="checkbox"/> None</p>	<p>Do you want to specify a default route IP address or allow Solaris installation program find one?</p> <p>The default route provides a bridge that forwards traffic between two physical networks. The IP address is a unique number that identifies each host on a network.</p> <p><input type="checkbox"/> Detect one upon reboot (Solaris installation program will detect an IP address)</p> <p><input type="checkbox"/> Specify one (you specify the IP address)</p> <p><input type="checkbox"/> None (if you do not have a router or do not want the system to detect an IP address).</p>	
<p><input type="checkbox"/> Time Zone</p>	<p>You will be asked to specify your default time zone. Selections are: Geographic Continent/Country/Region; Offset from GMT: and Time Zone File.</p>	
<p><input type="checkbox"/> Root Password</p>	<p>This is the root password for this system.</p>	
<p><input type="checkbox"/> Type of Installation</p>	<p>You will choose between an initial or upgrade installation. You need to choose initial if you are installing for the first time.</p>	
<p><input type="checkbox"/> Locale</p>	<p>This is the geographic regions where you want to install support.</p>	
<p><input type="checkbox"/> Disk Selection</p>	<p>This selects the disk on which you want to install Solaris software.</p>	

**Table 5. Description of Installation Prompts**

Prompt/Task	Description	Your Values
<input type="checkbox"/> Proxy Server Configuration	Do you have a direct connection to the Internet or do you need a proxy server to access to the Internet. If you use a proxy server, you must provide the host and port.	
<input type="checkbox"/> Software Group	What software groups do you want to install?  <input type="checkbox"/> Entire <input type="checkbox"/> Developer <input type="checkbox"/> Documentation <input type="checkbox"/> Java	
<input type="checkbox"/> Preserve Data	Allows you to preserve any data that exists on the disks where you are installing the Solaris software.	
<input type="checkbox"/> Layout File Systems Select one of the following:  <input type="checkbox"/> Auto Layout <input type="checkbox"/> Manual	Auto Layout allows the installation program to automatically lay out file systems on your disks. Manual allows you to customize your disk partitions.  <b>Select Manual.</b>	

## ■ Step 3: Install Solaris 10 Operation System for x86 Platforms

The following instructions apply only if you have purchased the ACSLS turnkey hardware platform. In this case, you receive the official Solaris distribution on DVD media.

For any other installations, such as downloading Solaris from the Sun website, you must refer to their Solaris installation manuals.

1. Insert the Solaris 10 Update -1 (January 2006 or later) Operating System for x86 into the DVD drive and power on the system.

The `Solaris Install Console` displays first. During the installation process, this console provides messages indicating what the system is doing.

2. Identify the console.

This is the type of console for the hardware version on which the Solaris OS is being installed. Use the arrow keys to select the version of the Solaris OS that matches the hardware configuration. For most x86 systems, this is the default selection (Solaris).

```
Solaris
Solaris Serial Console ttya (headless and rack-mounted
systems only)
Solaris Serial Console ttyb (for1x50, v60x and v65x)
```

3. Select the appropriate console and click <Next>.

The following prompt displays.

```
Booting 'Solaris'
kernel /boot/multiboot kernel/unix -B install_media=cdrom
Multiboot-elf...
```

Once loaded, six installation options are presented.

4. Select the type of installation to perform.

Default is **Solaris Interactive**. If you don't select anything within the time limit, the default is selected for you.

```
Select the type of installation you want to perform:

    1 Solaris Interactive
    2 Custom JumpStart
    3 Solaris Interactive Text (Desktop session)
    4 Solaris Interactive Text (Console session)

Enter the number of your choice followed by the <ENTER> key.
Alternatively, enter custom boot arguments directly.

If you wait for 30 seconds without typing anything,
an interactive installation will be started.

Select type of installation: 1

<<< starting interactive installation >>>

Booting kernel/unix...
```

When the first option is selected, or the counter times out, the following output is displayed. Keep in mind that the hardware information displayed,

such as "bge0", may differ from the exact data displayed below and that each step can take up to a couple of minutes to complete.

**Solaris Interactive**

```
Using install cd in /dev/dsk/c1t0d0p0
Using RPC Bootparams for network configuration information.
Attempting to configure interface bge0...
Skipped interface bge0...
Beginning system identification...
Searching for configuration files(s)...
Search complete.
```

**Proposed Window System Configuration for Installation:**

```
Video Devices: xxxxxxxxxxxxxxxx
Video Drivers: xxxxxxxxxxxxxxxx
Resolution/Colors: xxxxxxxxxxxxxxxx
Screen Size: xxxxxxxxxxxxxxxx
Monitor Type: xxxxxxxxxxxxxxxx
Keyboard Type: xxxxxxxxxxxxxxxx
Pointing Device: xxxxxxxxxxxxxxxx
```

```
Press <ENTER> to accept proposed configuration
or <ESC> to change proposed configuration
or <SPACE> to pause
```

```
<<< timeout in 30 seconds>>>
```

5. Select your system configuration.

A series of screens guide continue to guide you through the installation process.

6. Select a language for the system.

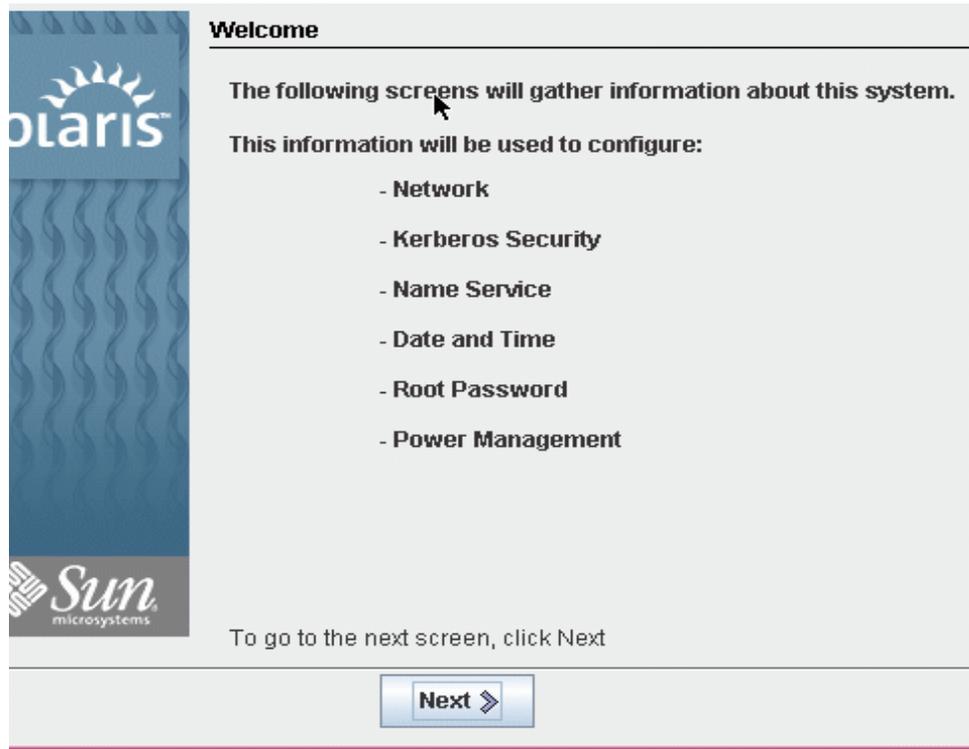
**Select Language**

0. English
1. French
2. German
3. Italian
4. Japanese
5. Korean
6. Simplified Chinese
7. Spanish
8. Swedish
9. Traditional Chinese

Please make a choice (0-9), or press h or ? for help:

After you have selected a language, the Solaris Install Console window displays. This window shows pertinent installation messages.

The Welcome screen then displays.



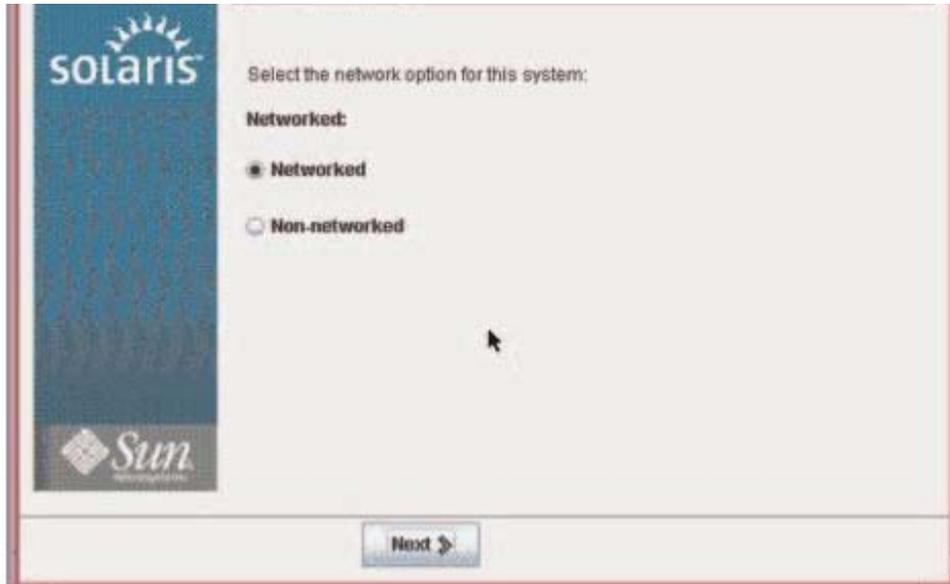
7. Click <Next> to continue the installation.

The configuration information you enter is checked by the system. If any information is found to be incorrect, you are prompted to correct it.

8. Select the Primary Network Interface and make sure it is connected to the network.

nge0  
bge0

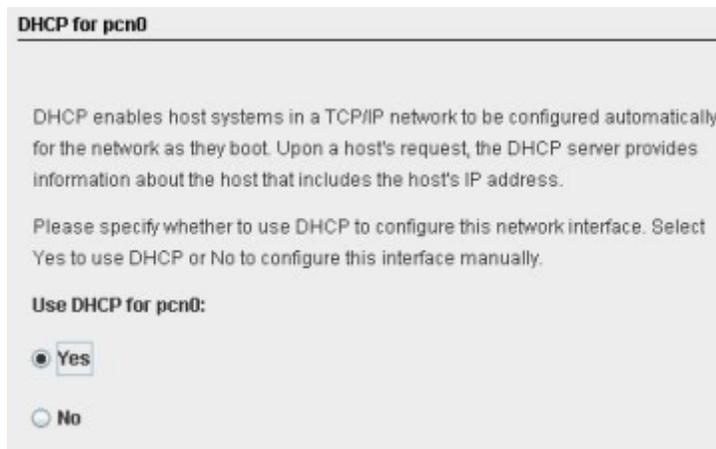
The Network Connectivity screen displays.



For most systems you can accept the default options.

9. Select **Networked** and click <Next>.

The DHCP for pcn0 screen displays.



10. Select **No**.

You are guided through several configuration screens. Refer to [Table 5 on page 77](#) for a description of these screen prompts. You can also use this table to enter your values beforehand.

After you have completed the configuration information, a screen displays with a summary of the configuration information entered.

**Confirm Information**  
**Selection Information**

```
Hostname: denver
IP Address: 193.168.1.1
System part of a subnet: Yes
Netmask: 255.255.255.0
Enable IPv6: No
Default Route: Autodetect Default
Name Service: None
Enable Kerberos: No
Timezone: US/Mountain
```

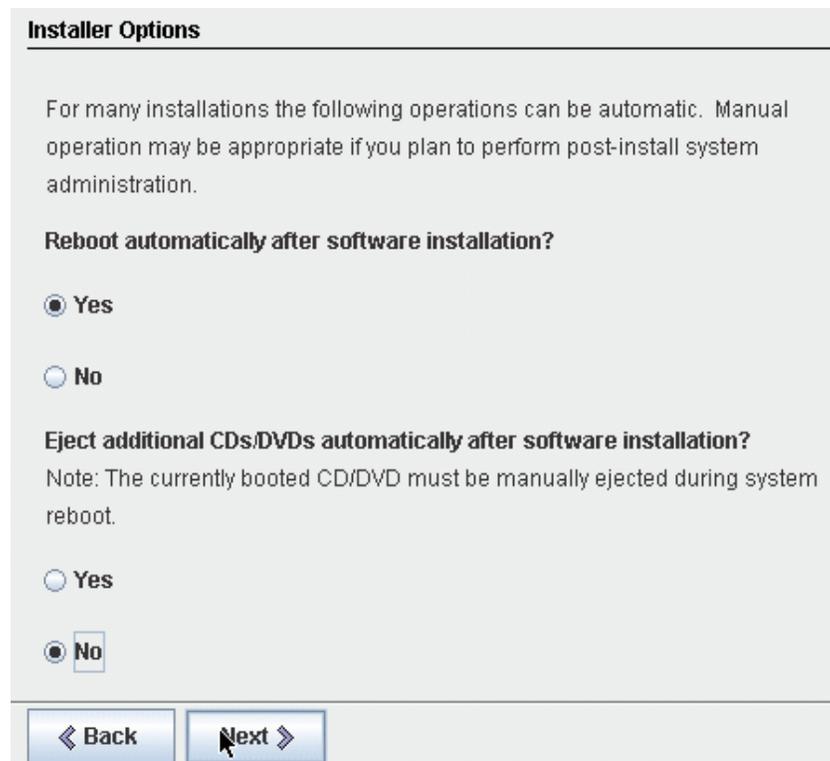
<Back Confirm>

**Note:** You may need to resize the window to view the entire configuration.

11. Verify all the information is correct and click **Confirm**.

Solaris installation is started. The Solaris Installation Welcome screen displays.

12. Click <Next> to begin Solaris installation.



13. Click **Yes** to both installer options if the system should reboot automatically at the end of the installation process, and if the DVD should be automatically ejected.

14. Click <Next>.

15. Click OK on the pop-up window to continue.

16. Select **CD/DVD** as your media type and click <Next>.

The system begins initialization. The License Agreement window displays.

17. Read the License Agreement and click Accept.

18. Click <Next>.

The select Upgrade or Initial Install window displays.

19. Select **Initial Install** and click <Next>.

The Select Type of Install window displays.

20. Select **Custom Install** and click <Next>.

21. Select your System Locale.

22. Select the software products needed and click <Next>.

You need to have these DVDs available to continue with the installation.

No additional software products are needed for ACSLS.

23. Select the **End User Group** option under Custom Packages and click <Next>.

24. Scroll down the list to select **Online Manual Pages** and click <Next>.

The Disk Selection window displays.

25. If the system contains more than one disk, select a disk from the list on the right hand side to remove the extra disk.

26. Click <Remove> to place that disk on the left hand side of the screen.

27. Click <Next>.

28. Select the disk for custom partitioning.

29. Click the box to place a check mark and click <Next>.

30. Select **Modify** and click <Next>.

The Select Disks for fdisk Partition Customization window displays. You need to configure multiple partitions on a single disk.

31. Click <Next>.

32. Accept the default (**No**) on the Preserve Data window and click <Next>.

33. Set the size of your Disk Partitions.

How you set up your disk partitions can have a significant impact on system performance. This is because disk I/O is used for file system operations and also by the virtual memory subsystem for paging and swapping.

**Note:**

- Assign the mount point for each of the partitions in the left column.
- Assign the size for each partition in the right column except for /export/backup.

You may be able to improve disk I/O performance by following the configuration guidelines described in the figures below.

34. Click <Modify> to layout the file system.

It is recommend that you have a minimum disk size of 36GB. The following guidelines assume a disk of 36GB or larger. The values defined here are more than adequate for each of the specified partitions.

**Figure 9. Primary Disk Partitions Guidelines**

Disk: c0t0d0		Disk Size: 36GB
		Minimum Requirements
0	/	5000 MB
1	swap	1000 MB
2	/var	5000 MB
3	/usr	5000 MB
4	/opt	5000 MB
5	/export/backup	5000 MB
6	/export/home	The remaining disk space should be assigned to this partition

35. Click Apply after entering the disk partitions.

36. Click OK.

The Lay Out File System refreshes with the values you entered.

**Note: Second disk support:**

If you have a second disk, scroll down and click on disk and select Modify. Follow the same procedures for partitioning as described above.

37. Verify the information and click <Next>.

The Ready to Install window displays.

38. Verify the information and click `Install Now`.

A series of screens display updating you on what is being installed.

After a few minutes the bar reaches 100 percent. The system asks if it should Pause or Reboot. Be sure the system reboots from the internal hard disk.

The GNU GRUB screen displays.

39. Highlight the Solaris option and press `Enter`.

The system reboots.

You are then asked if the NFS protocol should be overridden.

40. Accept the default (**No**).

This accepts the system's default NFS version 4 domain name. The boot process continues.

The Welcome screen displays.

41. Enter the `root` user name and password.

42. Click `<OK>`.

43. Continue with the installation by inserting the Software Companion disk.

44. **Reboot** after the last DVD is installed.

The Solaris 10 x86 is now installed. You are now ready to install ACSLS.

45. Select your desktop environment.

The Help Viewer window displays and the loading system information window displays. The Registration Wizard then displays.

46. Make your selection and click `<Next>`.

## ■ Step 4: Prepare for ACSLS Installation

Before you install ACSLS, complete the following steps:

1. If you need to apply any maintenance, refer to the installation instructions in the associated PUT or SPE or PTF document included with the DVD, or the document associated with the maintenance tar file on the Customer Resource Center (CRC) web page.
2. Verify that the server system hardware is properly configured, connected, powered on, and ready.
3. Connect the LMU to a valid serial port or network.

4. If your installation includes a SL8500 library or a 9300 library and TCP/IP LMU(s), connect the TCP/IP LMU(s) to the network used for LMU communication.
5. For SCSI-connected libraries make sure that your server is configured with an appropriate SCSI HBA.

Contemporary Sun Storagetek libraries are configured with LVC SCSI. Contemporary Sun x86 ACSLS servers support only LVD SCSI and single-ended library connections.

6. Make sure that each attached LMU and LSM is fully configured, powered on, and ready.

**Note:** The configuration script, `acsss_config`, will fail unless all LMUs and LSMs are fully configured, powered on, and ready.

7. If you have any communication problems refer to the troubleshooting chapter in the appendix.
8. ACSLS requires specific user IDs. If these user IDs are not defined before ACSLS installation, ACSLS creates them. When ACSLS installation creates the user IDs, the system assigns the user ID numbers and group ID numbers. To assign specific user ID and group ID numbers, you must define the following groups and users before installing ACSLS:

User	Group
<code>acsss</code>	<code>staff</code>
<code>acssa</code>	<code>staff</code>
<code>lib6</code>	<code>staff</code>
<code>informix</code>	<code>informix</code>

When these user IDs are defined before ACSLS installation, they can either be defined locally (on the ACSLS server) or via remote authentication methods (e.g., NIS or Kerberos). These user IDs must be defined with the following properties:

- The shell for these user IDs must be `sh` on Solaris. The exception is `acssa` (`ksh`).
- The home directories for the `acsss`, `acssa`, and `informix` user IDs must be under the ACSLS installation directory. By default, the ACSLS installation directory is `/export/home` (referred to as `$BASEDIR`). The home directories for the ACSLS user IDs are:

<code>acsss</code>	<code>\$BASEDIR/ACSSS</code>
<code>acssa</code>	<code>\$BASEDIR/ACSSA</code>
<code>informix</code>	<code>\$BASEDIR/informix</code>
<code>lib6</code>	<code>\$BASEDIR/ACSSS</code>

If the home directories for the ACSLS user IDs do not match their required locations, please either modify the home directories for these

users or delete the user IDs so they are added correctly during the ACSLS installation process.

## ■ Step 5: Install ACSLS

You begin ACSLS installation using `pkgadd`.

1. Log in as `root`.
2. Insert the ACSLS 7.1.1 CD.
3. In a terminal window or at the system prompt, enter

```
cd /cdrom/cdrom0
```

4. Begin installing using `pkgadd`:

```
pkgadd -d •
```

**Note:** Make sure you enter a space and a period after `-d`

`pkgadd` asks what package you want installed.

5. When prompted to select a package, select **STKacsls** and press **<Return>**.
6. Enter `y` or `n` at the prompt to install ACSLS in the default directory `/export/home/`.

Use the default directory `/export/home` or enter the directory where you want ACSLS installed.

7. Type `y` to at the prompt to install `setuid/setgid` files.
8. Select to continue at the super-user permission prompt.
9. Type `y` to at the prompt to install `STKacsls`.

User and group IDs are created (unless they already exist). Files being installed are displayed.

If the `acsss`, `acssa`, or `informix` user IDs are not defined with their home directories matching the ACSLS installation directory, the installation script displays a warning, for example:

```
***WARNING*** User acsss already exists, but its home
directory does not match the ACSLS installation
directory. Please change the acsss home directory to
/export/home/ACSSS after the installation.
```

If the `acsss`, `acssa`, or `informix` user IDs are created during installation, a default password is not created. You need to go into the **admintool** to create these passwords. In addition, they must be maintained or set to never expire.

10. Type `cd /`

11. Eject the CD.

**eject cdrom**

12. Optionally, apply any desired maintenance to ACSLS.

This ensures that any maintenance is applied before ACSLS is installed. To do this, you need to check the Customer Resource Center (CRC).

## Initiate the installation shell script

You now need to initiate the installation shell script.

1. Change directories:

```
cd /export/home/ACSSS/install
```

2. To initiate the installation shell script, enter

```
./install.sh
```

If shared memory settings have **not** been defined, you are prompted to allow the script to set shared memory and reboot the server:

```
This server is not set with shared memory required for
ACSLs and the Database.
```

```
Set shared memory and reboot the server to take effect at
kernel level? (y or n):
```

Respond `y` to the prompt.

The server reboots.

When the server comes back, log in as `root`, `cd` to `/export/home/ACSSS/install` (if you are not already in it), and restart `install.sh`.

3. Enter the database backup directory.

By default, this is `/export/backup`.

4. Press **<Enter>** to accept `/export/backup` as the default directory.

5. Respond `y` or `n` to the prompt for automatic startup on reboot.

**Option: If you have a SCSI or fibre-attached library continue with step 5.**

6. Respond (`y` or `n`) to the prompt for installing a SCSI device driver for SCSI libraries.

Do you want to install the scsi device driver for SCSI libraries? (y or n):

---

YES The following message displays:

Enter the SCSI device(s) that correspond to each library. Separate devices with a space (example: 4 5 6). Remember that SCSI devices are numbers between 0 and 15.

**Notes:**

- On most Solaris machines, target-7 is not a valid device address because target-7 is reserved for the SCSI initiator on the host-bus adapter.
- If you are not sure what numbers to enter in this step follow the procedures in [“Troubleshooting a SCSI Connection” on page 516.](#)
- Continue with Step 6.

---

NO

---

### Option: Configure second disk support

Upon completion of the library driver installation, the following message displays:

```
install.sh 1501: You can enable/disable second disk
support later as root by doing the UNIX command "cd /
export/home/ACSSS/install; ./sd_mgr.sh".
```

You can enable second disk support after you complete the procedures in this chapter. See [Appendix A, “Second Disk Support.”](#)

#### 7. Set the ACSLS user passwords.

These passwords must be maintained or set to never expire. You need to set the following password to prevent a security exposure: `acsss`, `acssa`, `informix`, and `lib6`.

You must set the passwords the first time you login to these IDs. To set the passwords:

- a. Ensure you are logged in as user `root`.
- b. Enter a password for each user ID.  
Example: `passwd acsss`
- c. Confirm the password at the prompt.

- d. Login to the user IDs you use.
  - e. Enter the password at the prompt.
8. If the `acsss`, `acssa`, or `informix` user IDs were not defined with their home directories matching the ACSLS installation directory, and the installation script displayed a warning, modify these user IDs so that their home directories are under the ACSLS base directory.

The following commands modify the above users' home directories. (You must be logged in as `root`.)

```
usermod -d $BASEDIR/ACSSS acsss
usermod -d $BASEDIR/ACSSA acssa
usermod -d $BASEDIR/informix informix
```

## ■ Step 6: Option: Import the Database

In the following cases, you need to import the data from a previous ACSLS release. They are:

- If you are migrating from a previous release.
- If you are re-installing ACSLS.

Refer to [“Importing the Database” on page 209](#).

## ■ Step 7: Configure your Library Hardware

**CAUTION:** If you imported data from a previous ACSLS release, you must start ACSLS and ensure all LSMs are online before configuring your library hardware. Then shut down ACSLS. This initializes the LSM types and protects your imported database information.

**Note:** You do not need to run `acsss_config` if you are not changing your library hardware. Go to [“Step 8: Enter your License Key Information.”](#).

You must run `acsss_config` to configure you libraries if:

- this is a new `acsss` installation,
- you are adding library hardware

Refer to [“Configuring or Updating your Library Hardware” on page 130](#).

## ■ Step 8: Enter your License Key Information

You must enter a license key to activate ACSLS. To avoid unnecessary delays obtain your license key before you begin the ACSLS installation.

**Note:** You have a 75-day trial period during which you do not have to enter a license key. Please enter your license key before the trial period expires.

Refer to [Chapter 2, “ACSLs License Key”](#) for procedures.

## ■ Step 9: Reset any Custom Dynamic or Static Variables

If you are migrating to ACSLS 7.1.1 from a previous release and have customized your dynamic or static variables, you need to reset them.

## ■ Step 10: Configure the Second Disk

After installing and configuring ACSLS on the primary disk, configure the second disk as described in [Appendix A, “Second Disk Support.”](#)

## ■ Step 11: Audit new libraries

You need to audit your libraries:

- If this is a new installation.
- If you are adding new libraries to an existing configuration.

## ■ Step 12: Verify ACSLS Installation

Use the following procedure to mount or dismount a cartridge to verify ACSLS.

### **Mount/dismount a cartridge to verify ACSLS:**

1. Verify you are logged in as `acsss`.
2. If ACSLS is not running, start it by entering:

```
rc.acsss
```

3. Query the server from the `cmd_proc` by entering:

```
query server
```

If messages are displayed indicating that the server is in recovery mode, wait for a message indicating that the server is running.

4. Verify that the following are online. You must have at least one of each online. If not, bring them online with the `vary` command.

```
query port all
```

```
query acs all
```

```
query lsm all
query drive all
```

5. Do you have at least one cartridge in an LSM?

---

YES Continue with the procedure after auditing or importing.

---

NO Enter a cartridge into an LSM.

---

6. Mount a cartridge by entering:

```
mount vol_id drive_id
```

**Hint:** Use the `query drive` command to get the ID of an available drive and the `query volume` command to get the ID of a library cartridge. See [Chapter 14, "Command References."](#)

7. Did you see a message indicating a successful mount?

A successful mount message is:

```
Mount: vol_id mounted on drive_id
```

---

YES Procedure is complete.

---

NO If an error message appears, run this verification procedure again, ensuring that you specified a valid, available drive and a library cartridge. If the mount/dismount still fails, call StorageTek for assistance.

---

8. Dismount the cartridge by entering:

```
dismount vol_id drive_id force
```

where *vol\_id* is the volume and *drive\_id* is the drive you specified in Step 6.

## ■ ACSLS Setup for Client Communication

To set up ACSLS for client communication, you include the client host name and IP address in `/etc/hosts` or in the NIS lookup table.

## ■ Uninstalling ACSLS 7.1.1

To uninstall ACSLS 7.1.1:

1. Log in as `acsss`.
2. Shut down ACSLS.

```
kill.acsss
```

3. Shut down the Informix database:

```
db_command stop
```

4. Remove pkgadd:

- Log in as root.
- Enter `pkgrm STKacsls`

5. De-install the primary disk:

- `cd /export/home`

If you installed ACSLS in another directory other than `/export/home`, `cd` to that directory.

- `rm -rf ACSSS informix`
- `cd /export/backup`

If you installed ACSLS backup in another directory other than `/export/backup`, `cd` to that directory.

- `rm -rf informix misc server_to_server`
- `cd /`
- `rm -rf INFORMIXTMP`

6. De-install the second disk if it was installed:

- `cd /second_disk`

If you installed the second disk in another directory other than `/second_disk`, `cd` to that directory.

- `rm -rf data`



# Installing ACSLS 7.1 on AIX

## 6

ACSLs 7.1 supports AIX 5.2 and AIX 5.3.

PUT0602A provides LPAR support for AIX 5.3, Maintenance Level 3. ACSLS must be the only application running in that particular LPAR.

To apply any STK ACSLS maintenance, refer to the installation instructions in the associated PUT or SPE or PTF document included with the CD, or the document associated with the maintenance tar file on the CRC.

The steps for installing ACSLS on all supported versions of AIX are:

1. Have your ACSLS license key information and ACSLS 7.1 CD available.
2. If you are upgrading from a previous version of ACSLS, export the database and store it on a server separate from the one on which you are installing AIX.
3. Prepare for AIX installation and install the AIX operating system.  
Please refer to the appropriate AIX Installation Guide.
4. Prepare for ACSLS installation.
5. Install ACSLS 7.1.
6. Import the database.  
**Note:** If you import the database, you can skip the next step.
7. Reset any custom dynamic or static variables. Performing an import and selecting option 4 merges any previous customized dynamic variables.
8. Configure your library hardware. (see [“Configuring or Updating your Library Hardware” on page 130](#)).
9. Enter your license key information (see [Chapter 2, “ACSLs License Key”](#)). A 75-day trial license is automatically installed when ACSLS is initially installed.
10. Audit new libraries.
11. Verify the ACSLS installation.

This chapter also discusses:

- ACSLS Setup for Client Communications
- Regressing to a Previous Version of ACSLS
- Uninstalling ACSLS 7.1

## ■ Step 1: Export the Database

If you are upgrading from a previous version of ACSLS, export the database and store it on a server separate from the one on which you are installing AIX. See [“Exporting the Database” on page 206](#).

## ■ Step 2: Prepare for AIX Installation

Before installing AIX, ensure you have met the minimum requirements. These requirements are:

**Figure 10. AIX Requirements**

Requirement	Description
Operating system	AIX 5.2 and AIX 5.3.
Memory requirements	256 MB minimum,
ACSAPI Clients must support ACSAPI packet version 3 or higher	<p>ACSLs supports these ACSAPI packet versions:</p> <ul style="list-style-type: none"> <li>• Packet version 3 (minimum)</li> <li>• Packet version 4 (recommended)</li> </ul> <p>The CSC Developer's Toolkit is used to create ACSAPI clients.</p> <p><b>Note:</b> The CSC Developer's Toolkit and ACSLS have supported packet version 3 since 1993.</p> <p>Support for versions 1 and 2 ended May 1, 2002.</p>
Disk requirements	<p>9 GB minimum</p> <p>You should reserve 3 GB for the ACSLS application and for backup.</p>
Disk partitions	<p>/export/home - 1 GB</p> <p>/export/backup - 2 GB or more</p>

**Figure 11. Client Setup Configuration Worksheet**

Installation Options	Valid Values	Your System Values
<input type="checkbox"/> Host Name		
<input type="checkbox"/> Internet Address		
<input type="checkbox"/> Network Mask		
<input type="checkbox"/> Gateway Address		

## ■ Step 3: Prepare for ACSLS Installation

Before you install ACSLS, complete the following steps:

1. Have your license key information and ACSLS 7.1 CD available.
2. Satisfy all requirement specified in the *ACSL S Product Information* guide.
3. Install the library hardware according to the Sun StorageTek installation instructions.



**CAUTION:** If you are installing a SCSI library, you *must* connect the server to the library before installing ACSLS.

4. Install the ACSLS server hardware according to the hardware manufacturer's instructions.
5. Verify that the server system hardware is properly configured, connected, powered on and ready.
6. If your installation includes a 9300 or 4400 library and a single or dual, serial LMU:
  - a. Validate that you have enough serial ports
  - b. Connect the serial LMU(s) to the appropriate serial ports on either the AIX server or the attached serial expander (Brick).
7. If your installation includes a SL8500 library or a 9300 library and TCP/IP LMU(s), connect the TCP/IP LMU(s) to the network used for LMU communication.
8. If your installation includes a SCSI library and a single-ended SCSI interface, do the following:
  - Verify proper single-ended SCSI cable terminator attachment. Make sure that the cable does not exceed 3 meters between the server and the SCSI library.

**Hint:** The single-ended configuration allows the LSM to connect to the host system bus. Noises on the system bus can corrupt

data bound for the system disk. Common sources of spurious noise should be avoided, including: hot plugging devices on the SCSI bus, attaching cables that are not actively terminated, and powering down an attached peripheral while UNIX is running.

- Verify the proper SCSI target address. Consult the appropriate SCSI hardware operator's guide for setting this address on the LSM.
- Set the SCSI library at SCSI target 0, target 2, or target 5. Typically, target 1 is reserved for a second disk, target 3 is reserved for a boot disk, target 4 is reserved for an external tape device, target 6 is reserved for a CD-ROM, and target 7 is reserved for a SCSI controller.

**Hint:** If 1,3,4,6, or 7 are already reserved on your system, use 0,2, or 5 for your SCSI library.

9. If you intend to use removable media for database backup, have a blank cartridge available for your backup device.
10. ACSLS requires specific user IDs. If these user IDs are not defined before ACSLS installation, ACSLS creates them. When the ACSLS installation creates the user IDs, the system assigns default user ID numbers and group ID numbers. To assign specific user ID and group ID numbers, you must create the following groups and users before installing ACSLS:

User	Group
acsss	staff
acssa	staff
informix	informix
lib6	staff

When these user IDs are created before ACSLS installation, they can either be created locally (on the ACSLS server) or via remote authentication methods (e.g., NIS or Kerberos). These user IDs must be defined with the following properties:

- The shell for these user IDs must be ksh on AIX.
- The home directories for the `acsss`, `acssa`, and `informix` user IDs must be under the ACSLS installation directory. The ACSLS installation directory is always `/export/home` on AIX. The home directories for the ACSLS user IDs are:

<code>acsss</code>	<code>/export/home/ACSSS</code>
<code>acssa</code>	<code>/export/home/ACSSA</code>
<code>informix</code>	<code>/export/home/informix</code>
<code>lib6</code>	<code>/export/home/ACSSS</code>

If the home directories for the ACSLS user IDs do not match their required locations, please either modify the home directories for these users or delete the user IDs so they are added correctly during the ACSLS installation process.

The following command creates the `acsss` user on AIX. (You must be logged in as `root`.)

```
mkuser -a home=/export/home/ACSSSpgrp=staff gecos="ACSL Control Login" acsss
```

The account information is:

```
acsss  home=/export/home/ACSSSpgrp=staff gecos="ACSL Control Login"
acssa  home=/export/home/ACSSA pgrp=staff gecos="ACSL SA Login"
informix home=/export/home/informix/IDS7.3pgrp=informix gecos="ACSL Informix
        Database Owner"
lib6   home=/export/home/ACSSSpgrp=staff
```

The following commands modify the `acsss`, `acssa`, and `informix` users' home directories. (You must be logged in as `root`.)

```
chuser -a home=/export/home/ACSSSpgrp=staff acsss
chuser -a home=/export/home/ACSSA pgrp=staff acssa
chuser -a home=/export/home/informix/IDS7.3 pgrp=informix informix
```

11. After creating the user ID accounts, you must set a password for all four accounts.

## ■ Step 4: Install ACSLS

### Verify Disk Space

Before installing ACSLS, verify that the hard disk was partitioned appropriately when the operating system was installed and configured.

### Verify disk partitions

1. At the system prompt enter `df -k` to verify the following:
  - The total KB space available is at least 1,00,000 for the `/export/home` file system for the database `/export/home` will contain the contents of the installation media.
  - The total KB space available for `/export/backup` is at least 2,00,000 for the database backup.

### Install ACSLS:

1. Is this a new installation?

---

YES    Go to Step 10.

---

NO     Make sure you exported the database.

Complete Steps 2 through 9 below.

Then continue with Step 10.

---

2. Log in as `acsss`.

3. Shut down ACSLS and the Informix database:

`idle` (from a `cmd_proc`)

`kill.acsss` (from a command prompt)

If you are upgrading from ACSLS 6.0 or later, enter the following commands:

`db_command stop`

`crontab -r` (removes crontab entries)

4. Log out as `acsss`.

5. Log in as `root`.

6. If your previous installation of ACSLS is release 6.1 or greater, remove the ACSLS package using SMIT. Otherwise, skip to Step 6.

- At the prompt, enter:

`smitty`

Press **<Enter>**.

- Move cursor to “Software Installation & Maintenance.” Press **<Enter>**.

- Move cursor to `Software Maintenance and Utilities` in the sub-menu. Press **<Enter>**.

- Move cursor to `Remove Installed Software` and, when prompted, enter:

`acsls.rte`

- Use the “down” arrow to move to the `Preview Only?` line. Use the tab to select the `No` option.

- Press **<Enter>**.

- Press **<Enter>**.

- Use the arrow key to scroll down to see whether the removal was successful, then press `[[F10]]` to exit.

7. Login as `root` and remove ACSLS, backup and other files:

For ACSLS 5.3 or 5.3.2, enter:

- `cd /export/home`
- `rm -rf ACSSS oracle`
- `cd /export/backup`
- `rm -rf oracle misc`

For ACSLS 5.4, enter:

- `cd /export/home`
- `rm -rf ACSSS sybase`
- `cd /export/backup`
- `rm -rf sybase misc`

For ACSLS 6.0 or 6.0.1, enter:

- `cd /export/home`
- `rm -rf ACSSS informix`
- `cd /export/backup`
- `rm -rf informix misc`

If the `server_to_server` directory exists, enter:

- `rm -rf informix misc server_to_server`
- `cd /`
- `rm -rf INFORMIXTMP nsr`

For ACSLS 6.1.0 or later enter:

- `cd /export/home`
- `rm -rf ACSSS ACSSA informix`
- `cd /export/backup`
- `rm -rf informix misc server_to_server`
- `cd /`
- `rm -rf INFORMIXTMP`
- `cd /var/tmp`
- `rm -rf acsls`



**CAUTION:** When removing files, especially in the root directory (`/`) be very careful to type commands exactly, including spaces only where shown.

8. Remove files under second disk (if installed)

- **`cd /second_disk`**

If you installed the second disk in another directory other than `/second_disk`, `cd` to that directory.

- `rm -rf data`

- `cd /second_disk/backup`

If you installed the second disk in another directory other than `/second_disk/backup`, `cd` to that directory.

For ACSLS 5.3 or 5.3.2, enter:

```
rm -rf oracle misc
```

For ACSLS 5.4, enter:

```
rm -rf sybase misc
```

For ACSLS 6.0 or later, enter:

```
rm -rf informix misc
```

9. Reboot to verify that no database processes are running before you begin the install.

```
reboot
```

10. Log in as `root`.

11. Insert the ACSLS CD into the CDROM drive.

```
mount -v cdrfs -o ro /dev/cd0 /cdrom
```

12. Install the ACSLS package using SMIT:

- At the prompt, enter:

```
smitty
```

- Select Software Installation and Maintenance
- Select Install and Update Software
- Select Install Software
- Enter

```
/cdrom
```

as the input device and press **<Enter>**.

- A new screen appears, asking for "SOFTWARE to install." Press **[[F4]]**.
- Arrow down to `acsls.rte` line and press **[[F7]]**.
- Press **<Enter>** three times.
- Wait for the OK prompt at the top of the screen, then press **[[F10]]** to exit.

13. Apply any desired maintenance to ACSLS.

This ensures that any maintenance is applied before ACSLS is installed.

## 14. To perform the installation:

- Change to the install directory. Enter:  
`cd /export/home/ACSSS/install`
- Initiate the installation shell script by entering:  
`./install.sh`
- Press **<Enter>** to accept the default directory `/export/backup` when you are prompted for the directory to use for database backups.
- Respond (y or n) to the prompt for automatic startup on reboot.

```
Do you want your system configured so that when it
reboots, it automatically restarts the ACSLS?
(y or n):
```

***Building the non-fibre SCSI mchanger Driver (Steps 15 through 21)***

**Note:** Do not perform these steps for fibre SCSI device drives. You have installed them in a previous step.

## 15. Respond to the prompt for installing a SCSI device driver for SCSI libraries.

```
Do you want to install the SCSI device driver for SCSI
libraries? (y or n):
```

---

```
YES Continue with the next step.
```

---

```
NO Go to Step 22.
```

---

## 16. Respond to the prompt for a SCSI or fibre adapter name.

```
Enter the SCSI adapter name you want configured. The
valid adapter names are:
```

```
The system displays SCSI adapter names.
```

## 17. At the prompt, type one of the SCSI adapter names displayed.

## 18. The screen will display a prompt similar to this example:

```
Use SCSI adapter "scsi0"?
```

```
Respond (y or n) to the prompt.
```

## 19. ACSLS will display the prompt:

```
Is this correct?
```

```
Respond (y or n) to the prompt.
```

The following message displays:

```
Enter the SCSI device(s) that correspond to each
library connected to SCSI bus <your-scsi-bus>. Separate
devices with a space (example: 4 5 6). Remember that
SCSI devices are numbers between 0 and 15.
```

**Note:** L180 and L700 LSMs can reside at any target from 0 to 15. 97xx LSMs will have a target address ranging from 0 to 6.

20. Enter a device number for each SCSI LSM.

The display will show all SCSI device numbers specified. The system then asks

```
Is this correct? (y or n):
```

Respond (y or n) as appropriate. The system displays a message similar to:

```
mchanger0 Available
```

**Hint:** If the status of the device is not shown as “Available,” refer to Troubleshooting Library Connections in [Appendix F, “Troubleshooting.”](#) Respond (y or n) to the prompt to configure additional SCSI libraries.

```
Do you want to configure additional SCSI libraries? (y
or n):
```

**Note:** If you install more than one library, the device numbers are displayed consecutively, starting with 0.

21. Remember to record the device name to use in configuring ACSLS.

22. Eject the CD:

```
umount /cdrom
```

23. Remove and store the ACSLS 7.1 CD-ROM .

24. Set the ACSLS user passwords.

You need to set the passwords for the following user IDs to prevent a security exposure: `acsss`, `acssa`, `informix`, and `lib6`.

You must set the passwords the first time you login to these IDs. To set the passwords:

- a. Ensure you are logged in as user `root`.
- b. Enter a password for each user ID.  
Example: `passwd acsss`
- c. Confirm the password at the prompt.

- d. Login to the user IDs you use.
  - e. Enter the password at the prompt.
25. If the `acsss`, `acssa`, or `informix` user IDs were not defined with their home directories matching the ACSLS installation directory, modify these user IDs so that their home directories are under the ACSLS base directory.

The following commands modify the `acsss`, `acssa`, and `informix` users' home directories. (You must be logged in as `root`.)

```
chuser -a home=/export/home/ACSSS acsss
chuser -a home=/export/home/ACSSA acssa
chuser -a home=/export/home/informix/IDS7.3 informix
```

## ■ Step 5: Import the Database

If you are:

- upgrading from a previous release of ACSLS, or
- re-installing ACSLS 7.1

you need to import the data from the previous installation. Refer to [Chapter 11, "Database Backup and Restore."](#)

## ■ Step 6: Configure your Library Hardware



**CAUTION:** If you imported data from a previous ACSLS release, you must start ACSLS and ensure all LSMs are online before configuring your library hardware. Then shut down ACSLS. This initializes the LSM types and protects your imported database information.

**Note:** You do not need to run `acsss_config` if you are not changing your library hardware. Go to ["Step 7: Enter your License Key Information" on page 110.](#)

You must run `acsss_config` to configure your libraries if:

- this is a new installation, or
- you are adding library hardware.

Refer to ["Configuring or Updating your Library Hardware" on page 130](#)

## ■ Step 7: Enter your License Key Information

You must enter a license key to activate ACSLS. To avoid unnecessary delays, obtain your license key before you begin the ACSLS installation. Refer to [Chapter 2, “ACSLs License Key.”](#)

**Note:** You have a 75-day trial period during which you do not have to enter a license key. Please enter your license key before the trial period expires.

## ■ Step 8: Reset any Custom Dynamic or Static Variables

If you are upgrading to ACSLS 7.1 from a previous release and if you have customized your dynamic or static variables, you need to reset them.

## ■ Step 9: Configure the Second Disk

After installing and configuring ACSLS on the primary disk, configure the second disk.

## ■ Step 10: Audit New Libraries

You need to audit your libraries:

- if this is a new installation, or
- if you are upgrading from an earlier release of ACSLS and you are adding new libraries.

## ■ Step 11: Verify ACSLS Installation

Use the following procedure to mount or dismount a cartridge to verify ACSLS.

### **Mount/dismount a cartridge to verify ACSLS:**

1. Verify you are logged in as acsss.
2. If ACSLS is not running, start it by entering  
`rc.acsss`
3. Query the server from the cmd\_proc by entering  
`query server`

If messages are displayed indicating that the server is in recovery mode, wait for a message indicating that the server is running.

4. Verify that the following are online. You must have at least one of each online. If not, bring them online with the vary command.

```
query port all
query acs all
query lsm all
query drive all
```

5. Do you have at least one cartridge in an LSM?

---

YES Continue with the procedure.

---

NO Enter a cartridge into an LSM.

---

6. Mount a cartridge by entering:

```
mount vol_id drive_id
```

**Hint:** Use the `query drive` command to get the ID of an available drive and the `query volume` command to get the ID of a library cartridge. See [Chapter 14, "Command References."](#)

7. Did you see a message indicating a successful mount?

A successful mount message is:

```
Mount: vol_id mounted on drive_id
```

---

YES Procedure is complete.

---

NO If an error message appears, run this verification procedure again, ensuring that you specified a valid, available drive and a library cartridge. If the mount/dismount still fails, call StorageTek for assistance.

---

8. Dismount the cartridge by entering:

```
dismount vol_id drive_id force
```

where *vol\_id* is the volume and *drive\_id* is the drive you specified in Step 6.

## ACSL S Setup for Client Communication

To set up ACSLS for client communication, you include the client host name and IP address in `/etc/hosts` or in the NIS lookup table.

## Regressing to a Previous Version of ACSLS

If you need to regress to a previous version of ACSLS after you install ACSLS 7.1, you need to perform the following steps:

- Export the database and save it on a tape or file external to the server.
- Uninstall ACSLS 7.1 as described in the steps below:
- Install the previous version of ACSLS

For assistance in regressing to a previous version, contact StorageTek Software Support. For more information, see *Requesting Help from Software Support*.

## Uninstalling ACSLS 7.1

To uninstall ACSLS 7.1:

1. Log in as `acsss`.
2. Shut down ACSLS.  
`kill.acsss`
3. Shut down the Informix database:  
`db_command stop`
4. Remove ACSLS 7.1 using the Package Manager:
  - Log in as `root`.
  - Enter the System Management Interface Tool (SMIT):  
`smit`

This brings up the System Management menu.

- **Select** Software Installation and Maintenance.
- **Select** Software Maintenance and Utilities.
- **Select** Remove Installed Software.
- **Enter the** SOFTWARE name:  
`acsls.rte`
- **Set** PREVIEW only? (remove operation will NOT occur)  
to `[no]`
- **Press <Enter>**.
- **Press <Enter>** again to confirm.

A series of messages showing the progress of the removal of ACSLS 7.1 will appear.

5. Perform a file cleanup:
  - `cd /export/home`
  - `rm -rf ACSSS informix`
  - `cd /export/backup`
  - `rm -rf informix misc server_to_server`
  - `cd /`
  - `rm -rf INFORMIXTMP nsr`
6. Perform a file cleanup for the second disk:
  - `cd /second_disk`
  - `rm -rf data`
  - `cd /second_disk/backup`
  - `rm -rf informix misc`
7. Reboot.



# ACSL S Configuration

# 7

---

ACSL S configuration is performed using the `acsss_config` menu.

## ■ Overview

You must use `acsss_config` to create your initial hardware configuration (option 8). After your initial library configuration, you can then use the `config` utility, to dynamically add, reconfigure, and remove ACSs, LSMs and tape drives and port connections to libraries. For more information and procedures, refer to [“Dynamic Configuration \(config\) utilities” on page 245](#).

### Note:

- `acsss_config` will fail unless all LMUs, LSMs, and transports are fully configured, powered on, and ready.
- Each rail on the SL8500 is configured as a separate LSM.
- Drive types and drive serial numbers are automatically updated when LSMs and/or drives are varied online. They are also updated by the `config` utility.

With `acsss_config`, you can change your dynamic and static variables (options 1-7), which includes:

- Setting the Client System Interface (CSI) variables
- Setting event logging
- Setting general product behavior, such as volume statistics and enabling auto-clean.
- Setting access control
- Setting automatic backup variables, such as how often you want to perform backups and retention periods.
- Rebuilding your access control tables
- Define event notification settings

**Note:** You need to reboot your system if you have set any static variable(s). You do not have to do this for the dynamic variables.

To save or view current variables:

- Display the current settings of the variables by entering:

**dv\_config -d**

- Save the current settings of the variables to a file by entering:

```
dv_config -d > filename
```

- View the file by entering:

```
vi filename
```

- Display values of dynamic options

```
dv_print
```

## ■ Accessing the ACSLS Feature Configuration Menu

When you install or upgrade ACSLS, system defaults have already been set based on most user environments. However, when necessary, you can use options 1-8 to change any of these settings. Options 1 through 7 allow you to change dynamic and static variables.

Use option 8 to add and define your library hardware. Use this option if you have just installed ACSLS for the first time, or, for example, when you need to add a new library.

To start `acsss_config`:

1. Exit out of the CDE until you reach the CDE login.
2. Log in as `acsss`.
3. At the password prompt, press `[[ENTER]]`.
4. Run the configuration script:

```
acsss_config
```

The ACSLS feature configuration screen appears.

**Note:** The configuration script, `acsss_config`, will fail unless all LMUs, LSMs, and transports are fully configured, powered on, and ready.

The following menu appears:

```

ACSLs feature configuration
Please enter the number followed by Return for your choice from the
following menu to configure product behavior in that area.
Press? followed by the Return key for help.
. 1: Set CSI tuning variables
. 2: Set event logging variables
. 3: Set general product behavior variables
. 4: Set access control variables
. 5: Set automatic backup parameters
. 6: Rebuild access control information
. 7: Event Notification settings
. 8: Define or Change Library Hardware Configuration
. E: Exit
Menu choice:

```

## ■ Verifying and Changing Dynamic and Static Variables

Options 1 through 7 allow you to change the following dynamic and static variables.

### Setting CSI Tuning Variables

The Client System Interface (CSI) handles communication between ACSLS and clients of other servers. It sets up how communications are handled for each client. If communication with one client is lost, the other clients are not affected and their communication continues without interruption. Multiple CSIs can run under ACSLS.

Option 1 allows you to set or change the following:

- **Prompt:** *Maximum age in seconds of pending requests in CSI request queue [172800]*

**Variable:** CSI\_CONNECT\_AGETIME

A dynamic variable, this setting determines how long ACSLS holds on to client requests to which it has not responded.

Valid entry is: 600 to 315360000 seconds. Default is 172800 seconds.

- **Prompt:** *Number of seconds between successive retries [4]*

**Variable:** CSI\_RETRY\_TIMEOUT

The default is 4 seconds.

A dynamic variable, this option specifies the minimum amount of time, in seconds, that the CSI should wait between attempts to establish a network connection.

You should modify this value if timing problems occur between the CSC and CSI.

- **Prompt:** *Number of retries for the CSI before a timeout condition occurs [5].*

**Variable:** CSI\_RETRY\_TRIES

A dynamic variable, this option specifies the number of attempts the CSI should make to transmit a message. Pending messages are discarded if a connection cannot be established within the number of retries specified. Default is 5 retries.

- **Prompt:** *Changes to alter use of the TCP protocol will not take effect until the product is restarted. CSI support for RPC using the TCP protocol is enabled [TRUE].*

**Variable:** CSI\_TCP\_RPCSERVICE

A static option, this option specifies if the CSI is to act as a TCP RPC server. Default is true.

- **Prompt:** *Changes to alter the use of the UDP protocol will not take effect until the product is restarted. CSI support for RPC using the UDP protocol is enabled [TRUE].*

**Variable:** CSI\_UDP\_RPCSERVICE

A static option, this option specifies if the CSI is to act as a UDP RPC server. Default is true.

- **Prompt:** *Changes to alter use of the port mapper will not take effect until the product is restarted. Enable port mapper: (ALWAYS / NEVER / IF\_DUAL\_LAN\_NOT\_ENABLED) [IF\_DUAL\_LAN\_NOT\_ENABLED].*

**Variable:** CSI\_USE\_PORTMAPPER

Default is IF\_DUAL\_LAN\_NOT\_ENABLED. A static option, the valid options are:

- ALWAYS - the port mapper should always be interrogated when the CSI is unable to send a message to a client.
- NEVER - the port mapper should never be interrogated when the CSI is unable to send a message to a client. Select this option if clients do not support a port mapper.

- `IF_DUAL_LAN_NOT_ENABLED` - the port mapper should be interrogated only if dual LAN support has not been enabled. If dual LAN support has been enabled, then it is assumed that clients do not support a port mapper.
- **Prompt:** *Number of ACSSURR persistent processes that should be started [1]:*

**Variable:** `SURROGATE_PROCESSES`

Valid entries: 0 or 1. Enter 0 if the LM Gateway is not installed.
- **Prompt:** *TCP/IP port number that the ACSLS surrogate (ACSSURR) socket will listen on for requests from a gateway system [50300].*

**Variable:** `SURROGATE_PORT`

This variable applies only to the Library Management (LM) Gateway. Valid entries: 50300 - 99999.
- **Prompt:** *Number of seconds to wait for data packets to be read on surrogate/gateway sockets:*

**Variable:** `SURROGATE_TIMEOUT`

This variable applies only to the Library Management (LM) Gateway. Valid entries: 1 - 600.
- **Prompt:** *Number of minutes to wait before deleting a stale queue entry [5].*

**Variable:** `SURROGATE_QUEUE_AGE`

This variable applies only to the Library Management (LM) Gateway. Valid entries: 5 - 60.
- **Prompt:** *Automatically start CSCI at ACSLS startup (TRUE/FALSE) [FALSE]:*

**Variable:** `START_CSCI_PROCESS`

This variable determines if the CSCI process is automatically started during ACSLS startup. The default value, FALSE, means that the CSCI is not started with ACSLS. Set this variable to TRUE to start the CSCI process with ACSLS.
- **Prompt:** *Enable CSI to be used behind a firewall (user-defined inbound port) (TRUE/FALSE) [FALSE]:*

**Variable:** `CSI_FIREWALL_SECURE`

A dynamic variable, this setting enables or disables the firewall security definition of a single, user-defined port for incoming requests to ACSLS. Values are:

**False** - select False if you do not want the ports on the ACSLS server to be restricted.

**True** - select True if you want the ACSLS server to operate behind a secured firewall.

**Prompt:** *Port number used by the CSI to receive incoming ACSLS requests.*

**Variable:** CSI\_INET\_PORT

This variable specifies the single, user-defined port used by the CSI for incoming TCP requests from one or more clients.

This variable is used only when the firewall secure CSI is enabled by setting CSI\_FIREWALL\_SECURE to True.

**Default:** 30031

**Valid entry:** 1024 - 65535

## Setting Event Logging Variables

Option 2 allows you to set or change the following event logs:

- **Prompt:** *Number of event log files to retain [9]:*

**Variable:** EVENT\_FILE\_NUMBER

9 is the default when ACSLS is installed or upgraded.

A dynamic value, this option allows you to specify the number of additional event log files to retain. If enabled and the current event log file size reaches the threshold size, the log is automatically copied to another file. Once the specified number of files is reached, the data in the oldest file is overlaid.

If you specify 0, ACSLS creates only one event log file with no additional event log files. When this file reaches its threshold size, an “Event log full” message is displayed periodically until the event log is renamed or deleted.

Specify a number between 1 and 9 to enable and specify the number of log files to retain. When the event log reaches the threshold size, the data is moved into the file `event0.log`. The next time the threshold size is reached, the `event0.log` data is moved to `event1.log` and the older event log data is moved to `event0.log`.

**Prompt:** *Changes to the logging directory will not take effect until the product is restarted. What directory should logging information be in [\$ACS\_HOME/log]:*

**Variable:** LOG\_PATH

This is the directory where the log files are to be placed. By default the ACSLS log files are placed in the \$ACS\_HOME/log directory. An alternative path can be used if there are disk space problems in the file system that contains \$ACS\_HOME.

The path must be an absolute path, that is, it must start with a / or \$ACS\_HOME.

Path length cannot exceed 120 alpha characters.

**Prompt:** *Maximum library server event log size in Kilobytes (1000 bytes) [500].*

**Variable:** LOG\_SIZE

This option specifies the threshold size for the Event Log in Kbytes (1,000 bytes). Default is 500.

Valid entry: 32-2147483

- **Prompt:** *Date/time format for all logs [%Y-%m-%d %H:%M:%S]:*

**Variable:** TIME\_FORMAT

This option specifies the format to use for printing the date and time information in the event and trace logs. The default format is %Y-%m-%d%H:%M:%S.

**Note:** This format does not apply to the backup dates presented for a point-in-time restore.

Press ? to get detailed information.

## Setting General Product Behavior Variables

Option 3 allows you to set or change the following:

- **Prompt:** *Enable Library Volume Statistics Gathering: (ON / OFF) [OFF]: ?*

**Variable:** LIB\_VOL\_STATS

This option specifies if the Library Volume Statistics (LVSTATS) information should be collected from the following operations: enter; eject; mount; dismount; and audit. Logging these statistics can take considerable disk space. Default is OFF.

- **Prompt:** *Number of acsss\_stats log files to retain [9]:*

**Variable:** VOL\_STATS\_FILE\_NUM

A dynamic value, this option allows you to specify file size and number of rollover files for the volume statistics log (acsss\_stats.log). Valid entry: 0-9. Default is 9.

9 is the default when ACSLS is installed or upgraded.

A dynamic value, this option allows you to specify the number of additional `acsss_stats` log files to retain. If enabled and the current `acsss_stats` log file size reaches the threshold size, the log is automatically copied to another file. Once the specified number of files is reached, the data in the oldest file is overlaid.

If you specify 0, ACSLS creates only one event log file with no additional event log files. When this file reaches its threshold size, an “`acsss_stats log full`” message is displayed periodically until the `acsss_stats` log is renamed or deleted.

Specify a number between 1 and 9 to enable and specify the number of log files to retain. When the `acsss_stats` log reaches the threshold size, the data is moved to the file `vol_stats0.log`. The next time the threshold size is reached, the `vol_stats0.log` data is moved to `vol_stats1.log` and the older event log data is moved to `vol_stats0.log`.

- **Prompt:** *Maximum library server access\_stats log size in Kbytes (1000 bytes) [500].*

**Variable:** VOL\_STATS\_FILE\_SIZE

This option specifies the threshold size for the `acsss_stats` log in Kbytes (1000 bytes). Default is 500.

Valid entry: 32-2147483

- **Prompt:** *Select cleaning cartridge ordering method [VOLID\_SORT]*

**Variable:** UNIFORM\_CLEAN\_USE

Valid selections are:

- **VOLID\_SORT** - orders the cleaning cartridges by the volume identifier. This uses up a cleaning cartridge before moving to the next one. If you select this option, ACSLS uses and returns cleaning cartridges in the same order as in previous releases of the ACSLS software. This is the default value when ACSLS is installed or updated.
  - **LEAST\_USED** - orders the cleaning cartridges by usage. If you select this option, ACSLS sorts the volume list in reverse order of usage and returns the cartridges with the fewest usages first. This spreads out usage of cleaning cartridges uniformly.
  - **MOST\_CAPACITY** - orders the cleaning cartridges by number of uses left. If you select this option, ACSLS sorts the volume list based on the number of uses left on the cleaning cartridge and returns the cartridges with the most uses left first. This uses up all cleaning cartridges close to the same time.
- **Prompt:** *Enable Transport Auto Clean Option (TRUE/FALSE) [TRUE]:*

**Variable:** AUTO\_CLEAN

Valid selections are:

- TRUE - when a transport requires cleaning, ACSLS automatically cleans the transport before the next mount.

**Note:** You cannot enable auto-cleaning for drives attached to SCSI-attached LSMs. ACSLS only allows you to clean these drives by manually mounting a cleaning cartridge. Use the LSM control panel to enable auto-cleaning.

- FALSE - does not enable automatic cleaning
- **Prompt:** *Library Server startup state (RUN/IDLE) [RUN]:*

**Variable:** AUTO\_START

This option specifies the initial ACSLS state after recovery completes during server software initiation.

Valid selections are:

- RUN - user requests are processed immediately after recovery completes.
- IDLE - prevents user requests from being immediately processed after recovery completes.

Setting this option to IDLE could be useful if you need to vary device states or perform some operational activity before you allow server access to your users.

- **Prompt:** *Changes to the number of mount processes ACSLS supports will not take effect until the product is restarted. Number of mount processes [2]:*

**Variable:** MAX\_ACSMT

Valid entry is 1 to 5.

StorageTek recommends that you accept the default value at initial configuration, then change the value as required. Increasing this value may improve performance.

- **Prompt:** *Enable QUERY persistent processes (TRUE/FALSE) [TRUE]:*

**Variable:** ENABLE\_ACSQY

This options allows you to execute queries as one or more persistent processes.

StorageTek recommends that you accept the default value at initial configuration, then change the value as required. Increasing this value may improve performance.

Valid selections are:



- TRUE - enables persistent query processes.

**CAUTION:** With a minimum configuration system, higher than 10 query persistent processes could use up system resources and not allow ACSLS to start up.

- FALSE - disables persistent query processes. Queries are invoked as request processes.
- **Prompt:** *Changes to the number of query processes ACSLS supports will not take effect until the product is restarted. Number of processes processes [2]:*

**Variable:** MAX\_ACSQY

This option specifies the number of persistent query processes to create, only if you enabled it in the above step. Valid numbers are 1 to 5.

StorageTek recommends that you accept the default value at initial configuration, then change the value as required. For large configurations (more than eight LSMs), this value may need to be increased to improve performance.



**CAUTION:** If this number is set too high, ACSLS may not be able to start up properly. Either lower the number or increase the maximum allowable process per user.

- **Prompt:** *Changes to the maximum number of ACSLS processes will not take effect until the product is restarted. Number of ACSLS processes [40]:*

**Variable:** MAX\_ACS\_PROCESSES

Valid numbers are: 32 to 10000.

A transient process is used to satisfy all requests except: mount; dismount; lock; unlock; clear\_lock; and query\_lock. The default, 40 processes, works for all ACSLS processes except extremely large configurations.



**CAUTION:** It is recommended you contact Software Support for advice before changing this value.

- **Prompt:** *Enable automatic enter tracing facility (TRUE/FALSE) [FALSE]:*

**Variable:** TRACE\_ENTER

This option specifies if the results of automatic enter operations are written in the event log.

Valid selections are:

- TRUE - enables messages to be written to the event log at the end of each automatic enter operation. This is the only method for determining why volumes were not entered during an automatic enter operation.

- FALSE - disables this feature and helps minimize the number of messages written to the event log.
- **Prompt:** *Enable volume tracing facility (TRUE/FALSE) [FALSE]:*

**Variable:** TRACE\_VOLUME

This option specifies if unsolicited messages are displayed when adding/deleting a volume(s) in the database.

Valid selections are:

- TRUE - enables unsolicited messages to be displayed whenever a volume is added or deleted in the database. Operations that can generate this messages are: audit; mount; dismount; enter; eject; recover; and vary.
- FALSE - disables this feature and helps minimize the number of messages written to the event log.
- **Prompt:** *Database isolation level. See Help for explanation of levels. [1]:*

**Variable:** ISOLATION\_LEVEL

The isolation level is used to define the degree of concurrence among processes that attempt to access the same rows simultaneously in Informix Database. The database isolation level affects read concurrency when rows are retrieved from the database as in queries and volume reports. The database server uses shared locks to support different levels of isolation among processes attempting to access data. The update or delete process always acquires an exclusive lock on the row that is being modified. The level of isolation does not interfere with rows that are being updated or deleted. If another process attempts to update or delete rows that are being read with an isolation level of Repeatable Read, that process is denied access to those rows.

You can choose from four isolation level options.

- **0 - Dirty Read**  
Provides zero isolation. Dirty Read is appropriate for static tables that are used for queries. With a Dirty Read isolation level, a query might return a phantom row, which is an uncommitted row that was inserted or modified within a transaction that has subsequently rolled back. No other isolation level allows access to a phantom row. Dirty Read is the only isolation level available to databases that do not have transactions.
- **1 - Committed Read (default isolation level)**  
Guarantees that every retrieved row is committed in the table at the time that the row is retrieved. Even so, no locks are acquired. After one process retrieves a row because no lock is held on the row, another process can acquire an exclusive lock on the same row and modify or

delete data in the row. Committed Read is the default level of isolation in a database with logging that is not ANSI compliant.

- **2 - Cursor Stability**

Acquires a shared lock on the selected row. Another process can also acquire a shared lock on the same row, but no process can acquire an exclusive lock to modify data in the row. When you fetch another row or close the cursor, the database server releases the shared lock. If you set the isolation level to Cursor Stability, but you are not using a transaction, the Cursor Stability isolation level acts like the Committed Read isolation level. Locks are acquired when the isolation level is set to Cursor Stability outside a transaction, but they are released immediately at the end of the statement that reads the row.

- **3 - Repeatable Read**

Acquires a shared lock on every row that is selected during the transaction. Another process can also acquire a shared lock on a selected row, but no other process can modify any selected row during your transaction or insert a row that meets the search criteria of your query during your transaction. If you repeat the query during the transaction, you reread the same information. The shared locks are released only when the transaction commits or rolls back. Repeatable Read is the default isolation level in an ANSI-compliant database.

- **Prompt:** *Number of days to retain volumes that are identified as ABSENT or EJECTED in the database. [5]:*

**Variable:** ABSENT\_VOLUME\_RETENTION\_PERIOD

A dynamic variable, it sets the number of days (0-999) to keep absent or ejected cartridges in the database. Unless you reenter the volume, ACSLS automatically deletes the volume from the database when the retention period expires.

If you never want the volume to be deleted, enter 999. To delete the volume immediately, enter 0.



**CAUTION:** Setting a high retention period can cause a large number of absent or ejected volumes to be retained. This uses more space in the database and causes it to fill up faster.

- **Prompt:** *Report ABSENT or EJECTED volumes to ACSAPI clients. [FALSE]:*

**Variable:** ENABLE\_STATUS\_VOLUME\_ABSENT

Valid selections are:

- TRUE - reports absent and ejected volumes to ACSAPI clients.
- FALSE - absent and ejected volumes are not reported to ACSAPI clients (as occurred in previous versions of ACSLS).

- **Prompt:** *Status of MISSING volumes reported to ACSAPI clients: TRUE=MISSING. FALSE=IN TRANSIT [FALSE]:*

**Variable:** ENABLE\_STATUS\_VOLUME\_MISSING

This option sets the status code to be returned to ACSAPI clients for missing volumes.

- TRUE - reports missing cartridges to ACSAPI clients.
- FALSE - missing volumes are not reported to ACSAPI clients (as occurred in previous versions of ACSLS).

## Setting Access Control Variables

Option 4 allows you to set or change the following access control variables:

- **Prompt:** *Changes to command access control will not take effect until the product is restarted. Access control is active for commands. (TRUE/FALSE) [FALSE].*

**Variable:** AC\_CMD\_ACCESS

Specifies if access control is active for commands. This allows control over which ACSLS command each network client is allowed to execute.

The default, false, disables access control.

- **Prompt:** *Default access for commands (ACCESS/NOACCESS) [ACCESS].*

**Variable:** AC\_CMD\_DEFAULT

Specifies the default access to commands if there is no relevant access control lists.

The default, ACCESS, allows command access unless otherwise denied through configuration information provided in the access control files.

- **Prompt:** *Changes to volume access control will not take effect until the product is restarted. Access control is active for volumes. (TRUE/FALSE) [FALSE].*

**Variable:** AC\_VOL\_ACCESS

Specifies if access control is active for volumes. This allows control over which ACSLS clients can access specific volumes. You must restart ACSLS for this option to take effect.

The default, false, disables access control for volumes.

- **Prompt:** *Default access for volumes (ACCESS/NOACCESS) [NOACCESS].*

**Variable:** AC\_VOL\_DEFAULT

Specifies the default access to volumes if there is no relevant access control lists.

The default, NOACCESS, disallows volume access unless otherwise granted through configuration information provided in the access control files.

- **Prompt:** *Messages will be logged when access to commands or volumes is denied (TRUE/FALSE) [FALSE].*

**Variable:** AC\_LOG\_ACCESS

Specifies if a message is logged in the event log when access to a command or volume is denied. The command or volume id is logged, as well as the access id and host id if this option is enabled.

## Setting Automatic Backup Variables

Option 5 allows you to set or change the following automatic backup variables:

- **Prompt:** *Would you like to modify the automatic backup settings? ( y or n)*

**Variable:** None. The automatic backup settings are saved in crontab. Please use acsss\_config to modify these settings.

Valid selections are:

- Y - If yes, you are asked if you want to back up everyday. If “yes”, then you must enter the time in HH:MM format.

If no, you must select the day(s) of the week to back up. Then you must enter the time in HH:MM format.

- N - you go to the next question regarding the retention period of SDM-managed database backup files.
- **Prompt:** *Set the number of days for the retention period of SDM-managed database backup files. See Help for explanation. [5].*

**Variable:** RETENTION\_PERIOD

The option sets how many days DSM tracks the data on your storage cartridges before the cartridge is designated as not eligible for reuse. Valid number of days: 4-30

The retention period is the period of time for which you retain backups. Transaction log files are *retained* with each backup, increasing the space needed in the backup partition. When choosing a retention period, consider the following conditions specific to your site:

- The size and activity of your library
- The number of automatic and manual backups performed in a given period of time

**Notes:**

When you run a manual backup with the `bdb.acsss` utility, an automatic backup to local disk is also done, thereby increasing the number of backups in the backup partition.

The retention period should be greater than the interval between backups. For example, if you have a retention period of four days, you should take automated or manual backups at least three times a week, with no more than three days between backups.

- The period of time for which you would like to retain backups and transaction log files, keeping in mind that long retention periods increase the space needed in the backup partition.

## Rebuilding Access Control Information

Option 6 rebuilds your access control tables, if access control is alive and/or has been changed.

## Defining Event Notification Settings

Option 7 allows you to define your event notification settings.

- **Prompt:** *Set the number of event messages sent to a client between two successive client checks by ACSLS. [100]*

**Variable:** `CLIENT_CHECK_MESSAGE_COUNT`

This option specifies the number of event notification messages to send a client and the minimum check interval (set with next question) before ACSLS checks to verify the registered client is still alive.

Valid entry: 10 - 1000 messages

- **Prompt:** *Enter the minimum interval between two successive client checks by ACSLS (in minutes). [30]*

**Variable:** `CLIENT_CHECK_MIN_INTERVAL`

ACSLs checks periodically if a registered client is still alive. This check is performed when the minimum interval and the event notification message count is exceeded.

Valid entry: 1 - 600 minutes

- **Prompt:** *Enter the time a client has to respond to a client check with a check registration request (in seconds) [30]:*

**Variable:** CLIENT\_CHECK\_RESPONSE\_TIME

When ACSLS checks periodically if a registered client is still alive, the client must respond with a check registration request within the specified response time. If the client does not respond within this time period, the client's Event Notification registration is cancelled.

Valid entry: 5 - 300 seconds

- **Prompt:** *Enter the polling interval for confirming Event Notification client registration (in minutes) [60]:*

**Variable:** CLIENT\_CHECK\_POLLING\_INTERVAL

ACSL S checks periodically if a registered client is still alive. At the polling interval specified, ACSLS sends a client check response to all clients who have not issued register, unregister, or check registration requests within this polling interval.

Valid entry: 10 - 1440 minutes

**WARNING:** Reboot Your System If You Have Set Any Static Variable(s)

You do not have to do this for the dynamic variables.

## Configuring or Updating your Library Hardware

Use Option 8 to configure your library hardware for the first time or to add new libraries or re-configure existing libraries.

The ACSLS database must be updated whenever the physical configuration of the library changes. ACSLS will not run if the configuration defined in its database does not match the configuration defined in the LMU.

Examples of configuration changes requiring an update to the database include:

- Adding or removing an ACS, LSM (including a SCSI-attached LSM such as the 9710), Pass-Thru Port (PTP), transport, or standby LMU

**CAUTION:** After you install a new ACS or LSM, make sure that each attached LMU and LSM is fully (hardware) configured, powered on, and ready before you reconfigure ACSLS. Otherwise, the `acsss_config` script fails.

- Adding or removing a port connection between the server system and the LMU

Use the Build/Verify Library Configuration routine to update the database with all configuration changes except adding or removing port connections. Use the Configure Library Communications routine to reconfigure ports. Use both

routines if you have added or removed an ACS. Both of these routines warn you that existing configuration data will be overwritten.

You should perform an audit after using the Build/Verify Library Configuration routine to reconfigure the library.

- **Prompt:** *Configure library communications? (y/n):*

Enter **y** to establish communications between the ACSLS server and LMUs (ports)

- **Prompt:** *Library server database exists and will be overwritten, continue? (y or n):* **y**

This prompt only displays if library communications have been previously established and there is an existing database.

Accepting the default (**y**) changes the configuration information only.

If you enter **n**, you are asked to

- **Prompt:** *Number of ACSs to be supported:*

Enter the number of ACSs (between 1 and 127) that will be supported at your site. You must answer at least 1 to this prompt. Hardware must be installed to support one or more ACSs.

**Note:** A pair of L700e libraries connected by a pass-thru-port counts as one ACS.

- **Prompt:** *Is ACS #n in a partitioned library? (y or n)*

Enter **y** to create a partitioned library. If you enter **y**, the following prompt displays.

- Prompt: *Enter the partition ID for ACS #n:*

This must match the partition ID (that you want to manage) with that on the SLConsole.

- **Prompt:** *Number of connections to ACS #n*

This message appears for each ACS configured and establishes the number of communication ports for each ACS. The connections are:

- SCSI connection
- Serial or TCP/IP connection between the ACSLS and the LMU. Two connections are recommended

The ACSs must be physically connected. You must enter a decimal number between 1 and 16.

**Note:** A pair of L700e SCSI libraries connected by a pass-thru-port has two connections - one for each L700e.

**CAUTION:** When you reconfigure multiple (existing) ACSs, be sure to specify them in the same order as your current ACSs.

**Ensure that all ports specified for an ACS are actually connected to the same ACS.**

- **Prompt:** *Device or host - ACS#n, connection #n:*

Enter the device or host for each connection.

**Sample serial device names for 4400 or 9300 libraries**

```
Device or host - ACS #0, device #0: /dev/ttya
Device or host - ACS #0, device #1: /dev/ttyb
```

To specify the connection to a TCP/IP attached LMU, enter one of the following:

- An IP address
- A hostname
- A fully qualified hostname

**Note:** If a host name is specified, this host name must be mapped to the IP address entered into the 9330 LMU. This host name-to-IP address mapping is site-specific. Typically, this is done by the /etc/hosts file, your Domain Name Server, NIS, or NIS+

**Sample TCP/IP device names for 9300 libraries**

```
Device or host - ACS #0, connection #0: hostname1
Device or host - ACS #0, connection #1: hostname2
Device or host - ACS #1, connection #0: fully_qualified_hostname
Device or host - ACS #2, connection #0: 192.168.174.31
```

**Note:** A connection to a 9330 ACS can be either serial or TCP/IP, but not both.

**Sample device name for SCSI Library**

```
Device or host - ACS #1, connection #0: /dev/mchanger2
```

**Sample device name for pair of L700e libraries**

```
Device or host - ACS #1, connection #0: /dev/mchanger2
Device or host - ACS #1, connection #1: /dev/mchanger3
```

**Note:** For Solaris only: On a Solaris system, the device name is assigned during ACSLS installation when the SCSI device number is entered.

- **Prompt:** *This step builds a database image of your complete library hardware configuration. Before proceeding, make sure that your library is completely configured, that all subsystems are functional and powered on. Build/Verify library configuration? (y or n):*

Enter **y**.

Before proceeding, make sure that your library is completely configured, that all subsystems are functional and powered on.

This step updates the database with all configuration changes except adding or removing port connections.

- **Prompt:** *Library server database exists and will be overwritten, continue? (y or n):*

Enter **y**. As the script builds the library configuration, it displays the following message of each panel in each LSM.

```
ACS # n, LSM # nn, PANEL # nn, created
```

The script also generates a library configuration report and appends it to the following file:

```
/export/home/ACSSS/log/acsss_config.log
```

- **Prompt:** *Configure client system interfaces? (y or n):*

This step applies to Solaris only.

Respond (y or n) when the system asks whether to configure client system interfaces.

Enter **n** unless you are using an ICL client with OSLAN protocol.

Entering **y** displays the following message:

```
CSI SELECTION TABLE
```

```
1) OSLAN CSI Not Selected
2) ONC/RPC CSI Always Selected
```

```
Do you want to change the CSI selection table (n):
```

If **y**, the following message appears:

```
Select OSLAN CSI (n):
```

```
ONC/RPC is always selected.
```

- **Prompt:** *Are you backing up the database to tape?*

Enter **y** if you want to backup the database to tape.

**Note:** At this point in the re-configuration, it is recommended that you back up the database to a tape that can be stored offsite and used, if needed, for disaster recovery of the database.

- **Prompt:** *Do you want to backup the database to tape? (y or n)*

- Enter **y** and insert a blank tape when the following message appears:

```
Prepare for database backup..  
Insert database backup tape
```

Press RETURN to continue.

The database backup procedure `bdb.acsss` automatically executes and captures the configuration data to enable database journaling.

When the database backup is completed, the following message displays:

```
Database backup successfully completed.
```

- Enter **n** if you do not want to back up the database to tape. You return to the main `acsss_config` menu

**CAUTION:** The tape drive may appear idle during the backup. It may not make any sounds or flash any lights during this time. **DO NOT** remove the tape from the drive until you see the following success message indicating the backup is complete:

1. Remove the tape and store it in a safe place.
2. Enter **E** to exit from the main `acsss_config` menu.
3. Enter your license key if:
  - this is a new installation.
  - you are upgrading to a new ACSLS release.
  - you have increased the number of storage cells in your existing configuration in excess of your current licensed capacity.
4. Run `rc.acsss` to start the server.

If the automatic startup on reboot option was enabled during ACSLS installation, ACSLS is initiated automatically when the server system is rebooted.

ACSLS is now ready for library operations. If you want to enter ACSLS commands, log in as `acssa`.

5. Perform an audit of the Library

An audit updates the ACSLS database to match the actual inventory of library cartridges.

ACSLs access control allows you to selectively limit ACSAPI client access to commands and/or volumes. The System Administrator using `cmd_proc` *always* has access to all commands. Access control affects client CSC requests only. The `cmd_proc` requests are not affected.

Access control allows you to:

- Set command access control

Command access control allows you to restrict a user's ability to issue commands (for example, `mount` or `eject`) for all volumes.

- Set volume access control

Volume access control allows you to logically partition volumes in ACSLS and restrict users access to specified volumes or volume ranges.

- Set volume ownership

This chapter discusses:

- [“How ACSLS Access Control Works” on page 135](#)
- [“Setting up Access Control Variables” on page 136](#)
- [“Access Control Scenarios” on page 152](#)
- [“Maintaining and Modifying ACSLS Access Control” on page 158](#)

## ■ How ACSLS Access Control Works

ACSLs access control is enabled by the access control static variable. For more information, refer to [“Setting Access Control Variables” on page 127](#).

Once set up and enabled, ACSLS access control uses a set of client identification files and a series of allow and/or disallow files that you created for managing command and/or volume access. These access control files are located in the `/export/home/ACSSS/data/external/access_control` directory.

Then, when ACSLS receives a request, it performs the following:

- Verifies whether the client that made the request is in one of the client identification files.

- If the requesting client *did not* pass a `user_ID` field in the request packet, ACSLS attempts to match the requestor's IP address with that in the `internet.addresses` table.

If a match is found, ACSLS updates the `user_ID` field in the request packet with the `user_ID` in the `internet.addresses` record for that IP address.

**Note:** Multiple user IDs can be associated with one IP address.

- If the requesting client *did* pass in a `user_ID` field in the request packet, ACSLS attempts to match both the IP address and the `user_ID` with those in the `internet.addresses` table.
- If the input internet address/`user_ID` pair doesn't match any entries in the `internet.addresses` table, a NULL string will be substituted for the `user_ID` when validated against the allow and disallow permission tables.

This ensures that users not in the `internet.addresses` table will be managed by the system access control user defaults and removes any possibility of an unauthorized user gaining access to or control of another user's data.

- Checks the allow and/or disallow files to determine if the user is authorized to issue the command against a volume. If the user is authorized, ACSLS accepts the command. If the user is not authorized, ACSLS rejects the command.

## ■ Setting up Access Control Variables

To set up access control, you need to perform the following (these tasks are discussed in detail in later sections):

1. Turn on access control by running `acsconfig`, option 4.
2. Create a client-identification file for the corresponding network protocol the client systems will use.
  - `internet.addresses` file for TCP/IP protocol networks
  - `adi.names` file for OSLAN protocol networks
  - `lu62.names` file for SNA protocol networks (available only for AIX platform for ACSLS)

**Note:** Use `vi` or another text program to create the access control files described here. Samples of these files, which you can copy, rename, and modify, reside in the `$ACS_HOME/data/external/access_control` directory.

3. If command access control is required, create the command allow and/or disallow files

4. If volume access control is required, create the user `allow` and/or `disallow` files.
5. Set the volume ownership of the volumes.

The volume ownership ID and access control user ID is compared when checking volume access.

**Note:** If access control is already enabled, simply select Option 6 from the `acsss_config` menu to rebuild the access control tables.

It is not necessary to restart ACSLS to rebuild the tables using Option 6.

6. Enable the ACSLS access control static variables using `acsss_config` and restart ACSLS.

This builds the access control files. See [“Enabling ACSLS Access Control” on page 151](#).

## Creating a client-identification file for the corresponding network protocol

To implement ACSLS access control, you must create a client-identification file in the `$ACS_HOME/data/external/access_control` directory. In this step you will create a client-identification file for the corresponding network protocol the client systems will use, including:

- `internet.addresses` file for TCP/IP protocol networks
- `adi.names` file for OSLAN protocol networks
- `lu62.names` file for SNA protocol networks (available only for AIX platform for ACSLS)

### Creating an `internet.addresses` File for TCP/IP Protocol Clients

To implement ACSLS access control, you must create a client-identification file in the `$ACS_HOME/data/external/ access_control` directory. For clients using TCP/IP protocol, you create an `internet.addresses` file.

Each line in the `internet.addresses` file consists of two fields. The first is the internet address of the client computer. The second is a `user_ID` to identify the client computer. This `user_ID` is not necessarily the system logon. It is determined by the client system.

[Table 9](#) shows a sample `internet.addresses` file. In this sample file, both `cray` and `test_group` are `user_IDs` for the client computer whose Internet address is `65.102.32.56`.

**Figure 12. Example of `internet.addresses` file**

```
# Filename: internet.addresses
#
# This file contains the internet addresses of all hosts for which you want to have
# the ACSLS software automatically fill in the User ID (also known as the AccessID)
# in the message header.
#
# Each line should consist of a "dotted quad" Internet address and an associated
# name, separated by spaces or tabs. Neither the address or the name may have
# embedded tabs or spaces. Anything to the right of the name is treated as a
# comment. Blank lines, and lines starting with # are ignored.
#
# Revision History:
# xx/xx/xx Name
#####
# IP Address      user ID      Comments
65.102.32.56     cray         The Cray downstairs
65.102.32.56     test_group   Test partition for Cray
```

## Creating an `adi.names` File for OSLAN Protocol Networks

To implement ACSLS access control, you must create a client-identification file in the `$ACS_HOME/data/external/access_control` directory. For clients using OSLAN protocol, you create the `adi.names` file.

Each line in the `adi.names` file consists of two fields. The first is the OSLAN host name of the client computer. The second is a `user_ID` to identify the client computer. This `user_ID` is not the system logon.

[Figure 13 on page 139](#) shows a sample `adi.names` file. In this sample file, `unisys` is the `user_ID` for the client computer whose OSLAN name is `SystemA`.

**Figure 13. Example of adi.names file**

```

# Filename: adi.names
#
# This file contains a list of ADI names for which you want to have the ACSLS
# software automatically fill in the User ID (also known as the Access ID) in
# the message header.
#
# Each line should consist of the client host name and an associated name
# separated by spaces or tabs. Note that both names may be the same.
#
# Neither the host name or the associated name may have embedded tabs or spaces.
# Anything to the right of the second name is treated as a comment.
# Blank lines, and lines starting with # are ignored.
#
# REVISION HISTORY:
# xx/xx/xx      NAME      DESCRIPTION OF CHANGE.
#
#####
# Client      user_ID      Comments
# SystemA     unisys        Unisys mainframe
# SystemB     seattle         Unisys server located in Seattle

```

## Creating an lu62.names File for SNA Protocol Clients

To implement ACSLS access control, you must create a client-identification file in the `$ACS_HOME/data/external/access_control` directory. For clients using SNA protocol, you create the `lu62.names` file.

### Notes:

The SNA protocol is only available for the AIX platform of ACSLS. The Solaris platform of ACSLS does not support SNA network protocol.

Each line in the `lu62.names` file consists of two fields. The first is the SNA host name of the client computer. This host name uses the “SNA\_domain\_name.Control\_Point” naming convention for SNA clients. The second is a `user_ID` to identify the client computer. This `user_ID` is not the system logon.

[Figure 14 on page 140](#) shows a sample `lu62.names` file. In this sample file, `as400` is the `user_ID` for the client computer whose SNA name is `SYSPLEX.PROD`.

**Figure 14. Example of u62.names file**

```

#
# Filename: lu62.names
#
# This file contains a list of SNA names for which you want to have the ACSLS
# software automatically fill in the User ID (also known as the Access ID) in
# the message header.
#
# Each line should consist of the client host name and an associated name
# separated by spaces or tabs. Note that both names may be the same.
#
# Client host name uses the "SNA_domain_name.Control_Point" naming
# convention for SNA clients.
#
# Neither the host name or the associated name may have embedded tabs or spaces.
# Anything to the right of the second name is treated as a comment.
# Blank lines, and lines starting with # are ignored.
#
# Revision History:
# xx/xx/xx   Name           Description of change.
#
#####
# SNA.Client      user_ID      Comments
SYSPLEX.PROD     as400        AS400 server
Dev.TEST2        mvs           OS/390 server

```

## Command Access File Types

There are four command access file types:

- `command.COMMAND.allow` where *COMMAND* is the command name and must be capitalized  
Example: `command.AUDIT.allow`
- `command.COMMAND.disallow` where *COMMAND* is the command name and must be capitalized  
Example: `command.MOUNT.disallow`
- `command.ALL.allow`
- `command.ALL.disallow`

## Creating Command Allow and/or Disallow Files

The Allow and Disallow file types allow you to specify whether or not clients have access to one or more commands in ACSLS. These files are created in the `$ACS_HOME/data/external/access_control` directory. To do this, you need to:

- Create a `command.ALL.allow` file to identify users who are allowed to access all commands

- Create a `command.ALL.disallow` file to identify users who are not allowed access to any commands
- Create a `command.COMMAND.allow` file to identify users who are allowed to access the specified command
- Create a `command.COMMAND.disallow` file to identify users who are not allowed to access the specified command.

You can use combinations of these files to tightly control access to commands. Refer to [“Access Control Scenarios” on page 152](#) for examples of how combinations of files enable better ACSLS access control.

**Notes:**

Command `allow` and `disallow` files are mutually exclusive. If both `allow` and `disallow` files exist for a given command, ACSLS references the `allow` file and disregards the `disallow` file. If neither `allow` nor `disallow` files exist, ACSLS uses the dynamic variables `AC_CMD_DEFAULT` and `AC_VOL_DEFAULT` to control command and volume access.

[Figure 15](#) shows an example `command.ALL.allow` file that allows client 1 access to all commands.

**Figure 15. Sample `command.ALL.allow` file**

```
# Filename: command.ALL.allow
#
# This file contains the user_IDs of all clients
# with access to all commands.
#
# Last modified: 92/08/25 S. Johnson File created

# user_ID      Comments
client_1      Payroll
```

**Note:** When creating `allow` and/or `disallow` files, any text following the `user_ID` (in this example, `Payroll`) is interpreted as a comment. Also, blank or empty `user_IDs` are not allowed; nor are `user_IDs` with leading, trailing, or embedded spaces.

A `command.ALL.disallow` file should have the same format as the `command.ALL.allow`.

Figure 16 shows an example `command.START.allow` file that restricts access to the `start` command so that only a single client (`user_ID=stlouis`) can start ACSLS.

**Figure 16. Sample `command.START.allow` file**

```
#   Filename: command.START.allow
#
#   This file contains the user_IDs of all clients allowed to
#   perform the START command.
#
#   Last modified: 92/08/25   Bill Johnson.   Added acct-ibm

#   user_ID           Comments
#   stlouis           The DEC machine in St. Louis
```

A `command.COMMAND.disallow` file should have the same format as the `command.COMMAND.allow`, but with the appropriate name (in this example, `command.START.disallow`).

## Command Access Control Reference

Command access control can apply to the following commands (Table 6):

**Table 6. Commands to Which Command Access Control Applies**

audit	delete pool	eject	mount	set_clean	vary
cancel	dismount	enter <sup>1</sup>	query	set_scratch	
clear_lock	dismount_force	idle	query_lock	start	
define_pool	display	lock	set_cap	unlock	

Note:

- enter<sup>1</sup> - Policies apply to virtual enter and manual enter, but not for automatic enter.
- mount<sup>2</sup> - Policies also apply to mount scratch and mount read only.

Use Table 7 as a quick reference for determining when command access is allowed. In the *Situation* column, *COMMAND* is the name of the command requested (for example, AUDIT).

**Table 7. Command Access Quick Reference**

Situation	Access Allowed	Access Denied
The user is the System Administrator	X	
The user_ID is listed in <code>command.COMMAND.allow</code>	X	
The user_ID is listed in <code>command.ALL.allow</code>	X	
Either <code>command.COMMAND.allow</code> , <code>command.ALL.allow</code> , or both exist, and the user_ID is not listed in either file		X

**Table 7. Command Access Quick Reference**

Situation	Access Allowed	Access Denied
Neither <code>command.COMMAND.allow</code> nor <code>command.ALL.allow</code> exist and the <code>user_ID</code> is listed in <code>command.COMMAND.disallow</code>		X
Neither <code>command.COMMAND.allow</code> nor <code>command.ALL.allow</code> exist and the <code>user_ID</code> is listed in <code>command.ALL.disallow</code>		X
Neither <code>command.COMMAND.allow</code> nor <code>command.ALL.allow</code> exist and the <code>user_ID</code> is not listed in either file.	X	
None of the following files exist: <code>command.COMMAND.allow</code> ; <code>command.ALL.allow</code> <code>command.COMMAND.disallow</code> ; <code>command.ALL.disallow</code> and the <code>AC_CMD_DEFAULT</code> variable is set to <code>ACCESS</code>	X	
None of the following files exist: <code>command.COMMAND.allow</code> ; <code>command.ALL.allow</code> <code>command.COMMAND.disallow</code> ; <code>command.ALL.disallow</code> and the <code>AC_CMD_DEFAULT</code> variable is set to <code>NO ACCESS</code>		X

## Volume Access Control

### Creating Users Allow and Disallow Files

The foundation of volume access control is that each volume is either owned or not owned as identified in the `ACSLD.ownership.assignments` file. Usually, clients can access only volumes not owned and volumes that they own; however, using volume access control, volume owners can allow or deny other clients access to their volumes.

`ACSLD` controls access to volumes by means of `allow` and/or `disallow` files that you create in the `$ACS_HOME/data/external/access_control` directory.

### Using Allow and Disallow Templates

Templates can be used as follows:

- `users.SAMPLE.allow`

Use the `users.SAMPLE.allow` file to copy and create a `users.ALL.allow` file to identify users who own certain volumes to perform all commands on those volumes.

- `users.SAMPLE.disallow`

Use the `users.SAMPLE.disallow` template to copy and create a `users.ALL.disallow` file to identify users who are not allowed access to certain volumes to perform any commands.

- `users.COMMAND.allow`

Use the `users.COMMAND.allow` template to create, for example, a `users.EJECT.allow` file to identify users who are allowed to access owned volumes for the specified command.

- `users.COMMAND.disallow`

Use the `users.COMMAND.disallow` template to create, for example, a `users.MOUNT.disallow` file to identify users who are not allowed to access owned volumes for the specified command.

You can use combinations of these files to tightly control command access to volumes. Refer to [“Access Control Scenarios” on page 152](#) for examples of how combinations of files enable ACSLS access control.

When ACSLS checks command access for a volume, it first compares the `owner_ID` with the `client user_ID`. If these IDs are the same, then the client is granted access. If these IDs are not the same, then ACSLS looks for relevant `allow` and/or `disallow` files. If it finds the client's `user_ID` in one of these files, it grants or denies command access accordingly.

### ***Creating a `users.ALL.allow` access file***

1. Copy the `users.SAMPLE.allow` file and rename the file to `users.ALL.allow`.
2. For each user create a list of users who are allowed access to the volumes owned by the specified owner.

**Note:** The `users.ALL.disallow` file uses the same procedure and format as the `users.ALL.allow` file.

**Figure 17. users.ALL.allow file**

```

#
# Filename: users.ALL.allow
#
# This file contains a list of users who are allowed access to volumes owned by the
# specified owner, for the command specified in the file name. Note that if there
# is a "disallow" file for the command, it will be ignored and this "allow" file
# will take precedence.

# Each line should consist of two or more words, separated by spaces or tabs. The
# first word is the owner ID, and the second and following words are the IDs of
# users who are allowed access to volumes owned by this owner.

# None of the IDs may have embedded blanks or tabs. Blank lines, and lines starting
# with # are ignored.
#
# Revision History:
# xx/xx/xx Name:          Description of change.
#####

The following is an example - please delete all these lines if using this file as
a template.

ownerid_1 userid_1 userid_2 userid_3
ownerid_1 userid_4 userid_5
ownerid_2 userid_3

```

***Creating a users.EJECT.allow access file***

1. Copy the users.COMMAND.allow file and rename the file to users.EJECT.allow.
2. Create a list of users who are allowed access to the command specified in the file name.

**Note:** The users.COMMAND.disallow file uses the same procedure and format as the users.COMMAND.allow.

Figure 18 shows a sample `users.EJECT.allow` file for the `eject` command.

**Figure 18. users.EJECT.allow file**

```
# Filename: users.EJECT.allow
#
# This file contains owner_IDs and the user_IDs of clients
# which may eject the owner's volumes. Lines are of the form:
# owner-id user-id1 user-id2 user-id3 ...
# For example:
# cray stlouis denver
#
# Last modified: 92/08/25Bill Johnson. Added cray/stlouis combination
# owner-id user-id1 user-id2 user-id3 user-id4
# cray acct-ibm stlouis denver convex
# cray acct-dec stlouis denver acct-ibm
```

### Volume Access Control Commands

Volume access control applies to the following commands (Table 8):

**Table 8. Commands to which volume access control applies**

dismount	lock	mount_readonly	set_clean	unlock
eject	mount	query_volume	event_notification	mount scratch
query scratch	query mount *	display	set_scratch	

Use Table 9 as a quick reference for determining when volume access is allowed. In the *Situation* column,

*COMMAND* is the name of the command requested (for example, AUDIT).

**Table 9. Volume Access Quick Reference**

Situation	Access Allowed	Access Denied
The specified volume is unowned	X	
The user is the owner of the volume	X	
The user is the System Administrator	X	
The owner/user is listed in <code>users.COMMAND.allow</code>	X	
The owner/user is listed in <code>users.ALL.allow</code>	X	
If <code>users.COMMAND.allow</code> and/or <code>users.ALL.allow</code> exist, and the owner/user combination doesn't exist in either file		X
Neither <code>users.COMMAND.allow</code> nor <code>users.ALL.allow</code> exist and the owner/user is listed in <code>users.COMMAND.disallow</code>		X

**Table 9. Volume Access Quick Reference**

Situation	Access Allowed	Access Denied
Neither <code>users.COMMAND.allow</code> nor <code>users.ALL.allow</code> exist and the owner/user is listed in <code>users.ALL.disallow</code>		X
Neither <code>users.COMMAND.allow</code> nor <code>users.ALL.allow</code> exist, and <code>users.COMMAND.disallow</code> and/or <code>users.ALL.disallow</code> exist but the owner/user is not listed in either file	X	
None of the following files exist: <code>users.COMMAND.allow</code> ; <code>users.ALL.allow</code> <code>users.COMMAND.disallow</code> ; <code>users.ALL.disallow</code> and the <code>AC_VOL_DEFAULT</code> variable is set to <code>ACCESS</code>	X	
None of the following files exist: <code>users.COMMAND.allow</code> ; <code>users.ALL.allow</code> <code>users.COMMAND.disallow</code> ; <code>users.ALL.disallow</code> and the <code>AC_VOL_DEFAULT</code> variable is set to <code>NO ACCESS</code>		X

## Setting Volume Ownership

The volume owner ID and access control user ID is compared when checking volume access. Volume ownership can be established using:

- The `set owner` command.  
For more information, see [“set owner” on page 372](#).
- The `watch_vols` utility  
For more information, see [“watch\\_vols” on page 286](#).
- The `ownership.assignments` file.

The commands listed in [Table 10 on page 147](#) create new volume records in the database. When these new volumes are added to the database, the `ownership.assignment` file determines the initial owner of the volume.

**Table 10. Commands for Setting Volume Ownership**

Command	Volumes Affected
<code>enter</code>	Volumes entered using the <code>enter</code> command
<code>enter automatic</code>	Volumes entered without an explicit command via a CAP whose <code>cap_mode</code> is <code>automatic</code>
<code>mount scratch</code>	Mounting a scratch volume by issuing the <code>mount scratch</code> command or a regular <code>mount</code> command on a volume that is a scratch volume
<code>set scratch</code>	Setting a volume to become a scratch volume

**Table 10. Commands for Setting Volume Ownership**

<b>Command</b>	<b>Volumes Affected</b>
<code>set scratch reset</code>	Resetting a scratch volume to become a data volume

The default ownership of volumes created when the commands in [Table 10](#) are issued is defined in the `ownership.assignments` file. [Figure 19 on page 149](#) shows a sample ACSLS `ownership.assignments` file that resides in `/home/ACSSS/data/access_control`.

The `ownership.assignment` file contains information about how to assign ownership of volumes that may need to have ownership assigned to them as shown in the following example.

**Figure 19. Example of ownership.assignment file**

```

# Filename: ownership.assignments

# This file contains policies that assign ownership to volumes when they are
# entered, mounted, or scratched. Commands that can assign ownership to volumes
# are:

# ENTER          - Entering a volume using the Enter command.
# ENTER_AUTOMATIC - Entering volumes simply by putting them in the CAP.
#
# MOUNT_SCRATCH  - Doing a Mount Scratch, or Mounting a volume that happens to be
#                  a scratch volume.
# SET_SCRATCH    - Setting a data volume to be a scratch volume.
# SET_SCRATCH_RESET -Setting a scratch volume to be a data volume.
#
# Each of these commands has either one or two keywords associated with it.
# Keywords may be uppercase, lowercase, or mixed.
# Valid keywords are:
#
# Owner_Default  = Default user - name is specified in the line starting with
#                  OWNER_DEFAULT. If this line is missing, the volume will be
#                  unowned.
# Requestor      = User_ID in request
# Same           = Same owner - don't change
# unowned        = Volume is unowned
#
# Customers Revision History:
# xx/xx/xx Name          Description of change.
#####
# The name following OWNER_DEFAULT will be the Owner ID of all volumes for commands
# specified with Owner_Default. If the name is missing, the volumes will be
# unowned.

OWNER_DEFAULT          SYSTEM
# These commands have one keyword associated with them.
# The keyword Requestor is invalid for ENTER_AUTOMATIC.

# ENTER                Owner_default
# ENTER_AUTOMATIC      Owner_default

# These commands have two codes. The first applies to owned volumes, and
# the second applies to unowned volumes.
# The keyword Owner_default is invalid for MOUNT_SCRATCH.

MOUNT_SCRATCH          Same Requestor
SET_SCRATCH            Same Requestor
SET_SCRATCH_RESET      Same Requestor

```

Data in an `ownership.assignments` file are pairs or triplets of words separated by spaces or tabs, and come in three forms: special pairs of words, pairs of words, and triplets of words.

- Special pairs of words assign a default owner for all unassigned volumes. The first word in the pair is `OWNER_DEFAULT`; the second word in the pair is the keyword that identifies the default owner.

The data line is an example of a *special pair of words*:

```
OWNER_DEFAULT SYSTEM
```

In this example, `SYSTEM` is the default owner.

- Pairs of words are associated with new volumes. New volumes can be entered using the `enter` and `enter automatic` commands. The first word in the pair is the name of the command; the second word in the pair is the keyword that assigns ownership of the volume.

In [Figure 19 on page 149](#) the following data line is an example of a *pair of words* form:

```
ENTER Unowned
```

- Triplets of words are found with commands (`mount scratch`, `set scratch`, and `set scratch reset`) that can be applied to existing volumes (either owned or unowned). The first word in the triplet is the name of the command; the second word is the keyword that assigns ownership of owned volumes; the third is the keyword that assigns ownership of unowned volumes.

In [Figure 19 on page 149](#), the following data line is an example of the *triplet of words* form:

```
MOUNT_SCRATCH Same Unowned
```

[Table 11](#) summarizes, for each of the commands that can set volume ownership, the number of ownership keywords accepted, valid keywords, and default keywords.

**Table 11. Commands and Valid Ownership Keywords**

Command	Number of Allowed Keywords (Valid Keywords)	Default Keyword
<code>enter</code> <sup>2</sup>	1 (same, owner_default, requestor <sup>1</sup> , unowned)	unowned
<code>enter automatic</code> <sup>2</sup>	1 (same, owner_default, unowned)	unowned
<code>mount scratch</code>	2 (same, requestor <sup>1</sup> , unowned)	same (owned volumes), unowned (unowned volumes)

**Table 11. Commands and Valid Ownership Keywords**

Command	Number of Allowed Keywords (Valid Keywords)	Default Keyword
set scratch	2 (same, owner_default, requestor <sup>1</sup> , unowned)	same (owned volumes), unowned (unowned volumes)
set scratch reset	2 (same, owner_default, requestor <sup>1</sup> , unowned)	same (owned volumes), unowned (unowned volumes)

1. If the keyword `requestor` is specified and the `user_ID` contains a null string, the volume is unowned.
2. For commands that create new volume entries, the keyword `same` results in volumes being unowned.

**Note:** Keywords in an `ownership.assignments` file are case-independent and can be abbreviated. Valid abbreviations include `S`, `Own`, `REQ`, `Un`. Also, you may use dashes in place of the underscores in this file.

## Access Control Logging

The dynamic variable `AC_LOG_ACCESS` specifies whether a message is logged in the event log when access to a command or a volume is denied.

## Enabling ACSLS Access Control

ACSLS access control is disabled by default. Enable it by setting static variables using `acsss_config` Option 4. You will then need to stop and restart ACSLS.

- To enable command access control, set the static variable `AC_CMD_ACCESS` to `TRUE`. (This variable defaults to `FALSE` if this variable is not set or has a value other than `TRUE`.)  
`AC_CMD_DEFAULT` sets default access for commands (`ACCESS/NOACCESS`)
- To enable volume access control, set the static variable `AC_VOL_ACCESS` to `TRUE`. (This variable defaults to `FALSE` if this variable is not set or has a value other than `TRUE`.)  
`AC_VOL_DEFAULT` sets default access to volumes (`ACCESS/NOACCESS`)

### Enabling ACSLS access control:

1. Run `acsss_config`.  
The ACSLS feature configuration menu displays.
2. Select option **4**.

See [“Setting Access Control Variables” on page 127](#) for detailed instructions.

3. Stop and restart ACSLS.

## ■ Access Control Scenarios

Use the following scenarios to understand how access control can fit your needs. Note, these scenarios were based upon the following assumptions:

- You allowed ACSLS to fill in the `user_id` field, which means access to commands and volumes is restricted on a client computer basis.
- You have five clients that talk to the library. Entries in your `internet.addresses` file look like [Figure 20](#):

**Figure 20. Sample `internet.addresses` file**

```
# Filename:  internet.addresses

# IP Address      user ID
123.45.9.65      client_1
55.22.61.5       client_2
76.24.52.1       client_3
14.117.82.74    client_4
104.77.23.32    client_5
```

## Command Access Control Scenarios

- For command access control scenarios, you've set up default access to commands using the `AC_CMD_DEFAULT` dynamic variable. Possible values are `ACCESS` and `NOACCESS`. The variable defaults to `ACCESS`.
  - If you want most clients to have access to most commands, set `AC_CMD_DEFAULT` to `ACCESS`.
  - If you do not want most clients to have access to most commands, set `AC_CMD_DEFAULT` to `NOACCESS`.
- For volume access control scenarios, you've set up default access to volumes using the `AC_VOL_DEFAULT` dynamic variable. Possible values are `ACCESS` and `NOACCESS`. The variable defaults to `NOACCESS`.
  - If, in general, you want to restrict volume access to the owner of a volume, set `AC_VOL_DEFAULT` to `NOACCESS` (the default).
  - If you want most clients to have access to volumes owned by clients other than themselves, set `AC_VOL_DEFAULT` to `ACCESS`.

## Command Access Control Scenario 1

In this scenario, you want to allow most clients access to most commands, but you want to allow only `client_3` access to the `start` command. To do this, verify that the `AC_CMD_DEFAULT` variable is set to `ACCESS` and create a `command.START.allow` file with `client_3` listed.

Your file will look like [Figure 21](#):

**Figure 21. Sample `command.START.allow` file**

```
# Filename: command.START.allow
#
# This file contains the user_IDS of all clients allowed to perform the
# START command.
#
# Last modified: 92/08/25 P Johnson File created
#
client_3  Smith-Jones accounting
```

## Command Access Control Scenario 2

In this scenario, you want to allow most clients access to most commands, but you want to prevent `client_4` and `client_5` from using the `eject` command. To do this, verify that the `AC_CMD_DEFAULT` variable is set to `ACCESS` and create a `command.EJECT.disallow` file with `client_4` and `client_5` listed.

Your file will look like [Figure 22](#):

**Figure 22. Sample `command.EJECT.disallow` file**

```
#      Filename: command.EJECT.disallow
#
#      This file contains the user_IDS of all clients
#      restricted from performing the EJECT command.
#
#      Last modified: 92/08/25 F Johnson File created
#
client_4  Personnel
client_5  Hawkins
```

**Note:** An alternative way to achieve the same result would be to create an allow file and give `client_1`, `client_2`, and `client_3` access to the `eject` command. However, since this allow file would require more entries, it would take more time to set up and maintain.

## Command Access Control Scenario 3

In this scenario, you want to allow `client_1` access to all commands; `client_2` and `client_3` access to `mount` and `dismount` commands only; and `client_4` and `client_5` access to `mount`, `dismount`, `enter`, and

`eject` commands. To do this, create a `command.ALL.allow` with `client_1` listed.

Your file will look like [Figure 23](#):

**Figure 23. Sample `command.ALL.allow` file**

```
#      Filename: command.ALL.allow
#
#      This file contains the user_IDs of all clients
#      with access to all commands.
#
#      Last modified: 92/08/25  Bill Johnson  File created
#
client_1 Payroll
```

Then create a series of allow files that list which clients can use the other specific commands.

Your allow files will have the following names and entries ([Figure 24](#)).

**Figure 24. allow Files Names and Entries**

```
command.MOUNT.allow
  client_2
  client_3
  client_4
  client_5
-----
command.DISMOUNT.allow
  client_2
  client_3
  client_4
  client_5
-----
command.ENTER.allow
  client_4
  client_5
-----
command.EJECT.allow
  client_4
  client_5
```

**Note:** In the above scenario, the `AC_CMD_DEFAULT` setting is irrelevant, since `command.ALL.allow` governs behavior for all commands. However, you may want to set `AC_VOL_DEFAULT` to `NOACCESS`.

## Volume Access Control Scenarios

### Volume Access Control Scenario 1

In this scenario, you want to restrict volume access to the owner of a volume only. To do this, verify that the `AC_VOL_DEFAULT` variable is set to `NOACCESS` and make sure that none of the following files exist.

- `users.ALL.allow` or `users.ALL.disallow`
- `users.COMMAND.allow` or `users.COMMAND.disallow`

Where `COMMAND` is any ACSLS command.

### Volume Access Control Scenario 2

In this scenario, you want to, in general, restrict volume access to the volume owner, but you want `client_1` and `client_2` to be able to mount and dismount volumes belonging to either `client_3` or `client_4`. To do this, verify that the `AC_VOL_DEFAULT` variable is set to `NOACCESS` and create a `users.MOUNT.allow` with `client_1` and `client_2` having mount access to volumes owned by `client_3` and `client_4`.

Your file will look like [Figure 25](#):

**Figure 25. Sample `users.MOUNT.allow` file**

```
# Filename: users.MOUNT.allow
#
# This file contains the user_IDs of all clients allowed to perform the
# MOUNT command on volumes owned by the specified client.
#
# Last modified: 92/08/25 Bill Johnson File created
#
client_3  client_1  client_2
client_4  client_1  client_2
```

Then create a `users.DISMOUNT.allow` file with the same entries, so that `client_1` and `client_2` have dismount access to volumes owned by `client_3` and `client_4`.

### Volume Access Control Scenario 3

In this scenario, you want to, in general, restrict volume access to the volume owner, but you want `client_5` to be able to access all volumes, regardless of command. You also want to allow `client_1` to use the `mount readonly` and `dismount` commands on volumes belonging to `client_4`. To do this, create a `users.ALL.allow` with `client_5` having access to all commands for all other clients' volumes.

Your file will look like [Figure 26 on page 156](#):

**Figure 26. Sample users.ALL.allow file**

```
# Filename: users.ALL.allow
#
# This file contains the user_IDs of those clients having access to all
# volumes owned by the specified clients.
#
# Last modified: 92/08/25 Bill Johnson File created
#
client_1  client_5
client_2  client_5
client_3  client_5
client_4  client_5
```

Then create a `users.MOUNT_READONLY.allow` file with `client_1` having `MOUNT_READONLY` access to volumes owned by `client_4`.

Your file will look like [Figure 27](#).

**Figure 27. Sample users.MOUNT\_READONLY.allow file**

```
# Filename: users.MOUNT_READONLY.allow
#
# This file contains the user_IDs of all clients allowed to perform the
# MOUNT_READONLY command on volumes owned by the specified client.
#
# Last modified: 92/08/25 Bill Johnson File created
#
client_4  client_1
```

Finally, create a `users.DISMOUNT.allow` file with the same entry, so that `client_1` has `DISMOUNT` access to volumes owned by `client_4`.

**Note:** In the above scenario, the `AC_VOL_DEFAULT` setting is irrelevant, since `users.ALL.allow` governs behavior for all commands. However, you may want to set `AC_VOL_DEFAULT` to `NOACCESS`.

## Volume Access Control Scenario 4

In this scenario, you want to, in general, restrict volume access to the volume owner, but you want to allow all clients except `client_5` to access volumes owned by `client_2`. However, you don't want these clients to use the `eject` command. To do this, verify that the `AC_VOL_DEFAULT` variable is set to `NOACCESS` and create a series of `users.COMMAND.allow` files showing

```
client_2  client_1  client_3  client_4
```

for each command except the `eject` command.

Your files will look like [Figure 28 on page 157](#):

**Figure 28. Sample users.COMMAND.allow file**

```
# Filename: users.MOUNT_READONLY.allow
#
# This file contains the user_IDs of all clients allowed to perform the
# MOUNT_READONLY command on volumes owned by the specified client.
#
# Last modified: 92/08/25 Bill Johnson File created
#
client_2  client_1  client_3  client_4
```

**Note:** For the conditions set out in this scenario, creating a users.ALL.allow would not work because it leaves you no way to restrict access to the `eject` command.

## Volume Access Control Scenario 5

In this scenario, you want `client_1` to have access to a subset of volumes owned by `client_2`. To accomplish this, you need to set up a “virtual owner”, `client_3`, to own the subset of `client_2`'s volumes. Then, give `client_1` and `client_2` access to those volumes owned by `client_3`.

`client_3` is referred to as a “virtual owner” because `client_3` does not exist as a host or a real user. The sole purpose of `client_3` is to allow other “real” clients or users to share subsets of volumes.

To do this, verify that the `AC_VOL_DEFAULT` variable is set to `NOACCESS` and create a `users.ALL.allow` file with `client_1` and `client_2` having access to `client_3`'s volumes.

Your file will look like [Figure 29](#).

**Figure 29. Sample users.ALL.allow file**

```
# Filename: users.ALL.allow
#
# This file contains the user_IDs of those clients having access to
# all volumes owned by the specified clients.
#
# Last modified: 92/08/25 Bill Johnson File created
#
client_3  client_1  client_2
```

## ■ Maintaining and Modifying ACSLS Access Control

Once ACSLS access control is set up, you may need to perform the following tasks to maintain and modify it.

- Logging access control messages in the event log.
- Modifying or adding access control files, including the `internet.addresses`, `adi.names`, and `lu62.names` files and the `allow`, and `disallow` files.
- Reinitializing the access control information.
- Reporting on volume ownership.

### Logging Access Control Messages

The dynamic variable `AC_LOG_ACCESS` specifies whether a message is logged in the event log when access to a command or a volume is denied. By default, `AC_LOG_ACCESS` is set to `False`. Set it to `True` to enable this function.

### Reporting on Volume Ownership

`query volume` does not report on volume ownership. The only way to discover the owners of volumes is to run `volrpt` with the `-f` and `-z` options, as described below:

- `-f` specifies that the contents of `filename` be used to create a customized `volrpt`

**Note:** If the `-f` option is used, `filename` can be a dash (`-`), which specifies that the input will come from `stdin`. Input from `stdin` is terminated with `<CTRL>+D`.
- `-z` causes the `cell_id` (`acs`, `lsm`, `panel`, `row`, `column`) and `drive_id` (`acs`, `lsm`, `panel`, `drive`) fields to be zero-filled

Figure 30 on page 159 contains a sample file for use with the `-f` option. A similar sample file, `owner_id.volrpt`, is provided in the following directory:

```
$ACS_HOME/data/external/volrpt
```

The sample file in Figure 30 on page 159 shows the valid field names (which must be spelled exactly as shown in uppercase), the default field lengths, and the default delimiter length:

**Figure 30. Sample owner\_id.volrpt File**

```
#####
#
# File name: owner_id.volrpt
#
# This file describes the report layout for volrpt invoked with the -f option.
#   volrpt -f <filename>
#
# The format of a line is:
#   field_name  field_length  delimiter_length
#
# The field length is the number of characters which will be printed for
# the field. The delimiter length is the number of spaces that will be
# printed after the field. If you leave out the lengths, or specify a
# value of -1, the default values will be used. Default delimiters are
# always 2. Here are the fields and their default lengths.
#
# ACCESS_COUNT 5      2
# ACCESS_DATE 15     2
# CELL_ID 13        2
# DRIVE_ID 10       2
# ENTRY_DATE 15     2
# LABEL_ATTR 5      2
# LOCK_ID 5         2
# LOCK_TIME 15     2
# MAX_USE 5         2
# MEDIA_TYPE 7      2
# OWNER_ID 20       2
# POOL_ID 5         2
# VOLUME_ID 62      2
# VOL_STATUS 17     2
# VOLUME_TYPE 4     2
#
# Revision History:
# xx/xx/xx  NameChanges
#
#####
VOLUME_ID 6      2
VOL_STATUS 17   2
DRIVE_ID 12     2
CELL_ID      15   2
OWNER_ID -1    0
```

The input file consists of multiple lines, one line for each field to be printed on the report line. Fields are printed across the report line in the order they appear in the file. There is no attempt made to restrict fields to fit within an 80 character line, so users can put as many fields as they wish on a line. A line consists of:

```
field_name  field_length  delimiter_length
```

Where `field_length` is the number of characters to print for the field and `field_delimiter` is the number of spaces to print after the field. Lines starting with `#` are comment lines, and blank lines are ignored.

If a `field_length` or `delimiter_length` is not specified, the default values are used. These values are shown in [Figure 30 on page 159](#). Any field or delimiter length can be specified as `-1` which means use the default value. This is useful if the user wants to use the default field length, but wants to change the delimiter length. Thus the last field in the file might be specified as `-1 0` to get the default field length and to print out no spaces after the field.

If numeric fields (`pool`, `pool_id`, `lock_id`, `maximum use`, `access count`) or identifier (`cell`, `drive`) are too long for the specified format, asterisks are printed instead. All other fields are truncated if the format is shorter than the length needed to show the field.

[Figure 31](#) contains a sample volume report generated using the `-z` and `-f` options with `owner_id.volrpt` as the input file.

**Figure 31. Sample owner\_id.volrpt Using -f Option**

```
1998-06-30 14:02:18

TOTAL VOLUMES: 2 SEQUENCE: sort by volume identifier

Volume Volume                Home      Owner
Label  Status                    Drive ID  Location  ID
RB1400 VOLUME_HOME                Not-in-drv 0,1,1,0,0  cray
RB1401 VOLUME_IN_DRIVE           0,0,1,0   0,1,2,0,0  cray
```

## Error Conditions

The following are common error conditions you might encounter when creating the `volrpt`.

- If the file specified with the `-f` option is not found, `volrpt` prints a message to `stderr` and exits.
- If more than one `-f` option is specified, `volrpt` prints a message to `stderr` and exits.
- If a field name is misspelled, field values are non-numeric, or field values are less than `-1` or greater than `255`, a message is printed to `stderr` and the line in error is ignored.
- If a field is specified on more than one line, the second and subsequent lines are ignored and a message is printed to `stderr`. This means that duplicate fields are ignored.

## Modifying or Adding Access Control Files

To maintain ACSLS access control, you'll probably need to modify or add new client-identification files, including the:

- `internet.addresses` file for TCP/IP protocol clients
- `adi.names` file for OSLAN protocol clients
- `lu62.names` file for SNA protocol clients. (AIX platform for ACSLS only)

You will also modify or add new allow or disallow files.

The existing ACSLS access control files are located in the `$ACS_HOME/data/external/access_control` directory. To modify these files, use `vi` or another text program to make the required changes. Then save the files and re-initialize access control.

**Note:** If you have questions about the format of the information in the client identification files or the `allow`, and `disallow` files, see Option 4.

To add new files, copy, rename, and modify the appropriate sample files, which reside in the `$ACS_HOME/data/external/access_control` directory. Then save the files and re-initialize access control.

## Rebuilding Access Control Information

During configuration, ACSLS reads the access control information from the `internet.addresses`, `adi.names`, and `lu62.names` files, and the `allow`, `disallow`, and `ownership.assignment` files. ACSLS does not dynamically update the configuration when one of these files changes or a new file is added. In order to re-initialize the information, you must rebuild the access control information by running the `acsss_config` program, Option 6.

To rebuild the information, select the `Rebuild Access Control information` option through the `acsss_config` configuration script.

1. Run `acsss_config`.

The ACSLS feature configuration menu displays.

2. Select option **6**.

The system dynamically rebuilds the ACSLS access control information and then the ACSLS feature configuration menu reappears.

**Note:** You do not need to stop and restart ACSLS.



Managing your library(ies) may consist of the following tasks:

- [“Auditing the Library”](#)
- [“Adding a SCSI or fibre-attached library to ACSLS” on page 165](#)
- [“Creating an mchanger Device Node” on page 166](#)
- [“Removing an mchanger Device Node” on page 166](#)
- [“Using the Extended Store Feature” on page 167](#)
- [“Managing a Mixed-Media Library” on page 169](#)
- [“Managing a Dual-LMU Configuration” on page 175](#)
- [“Managing a Dual-LAN Client Configuration” on page 177](#)
- [“Registering the IP Addresses of Primary and Secondary LANs” on page 178](#)
- [“Setting the TCP/IP Connection Timeout Interval” on page 179](#)
- [“Troubleshooting Library Connections” on page 509](#)
- [“Registering for Event Notification” on page 180](#)

## ■ Auditing the Library

An audit updates the ACSLS database to match the actual inventory of library cartridges.

### When to Run an Audit

You run an audit to:

- Create volume information in the database if you do not enter cartridges through the CAP.

Example: you added an LSM to your library, opened the LSM door, and manually added cartridges to the LSM.

- Resolve discrepancies between the library and the database.

Example: run an audit if you opened an LSM door and manually removed cartridges instead of ejecting them through the CAP.

The audit either marks the volume absent or deletes information for the removed volumes from the database.

An audit ejects errant volumes and either marks the volume absent or deletes their information from the database. An errant volume has:

- An external label that duplicates one already scanned.
- A missing or unreadable external label and no virtual label.
- An invalid media type.
- An invalid volume ID.

ACSLs records any database changes from the audit in the event log and also displays cmd\_proc messages during the audit. Audits only apply to LSM storage cells, not to transports or CAPs. For information about running an audit, see command, [“audit” on page 294](#).

## Audit Intervals

Audit intervals depend on several factors, including your ACSLS configuration, your library configuration, the number of database changes required, and the scope of the audit. [Table 12](#) describes how the differences in the scope of an audit affect the audit intervals.

**Table 12. How the Scope of an Audit Affects the Audit Interval**

This audit takes less time than...	This audit
A diagnostic ACS/LSM	An online ACS/LSM
An ACS/LSM dedicated to the audit	An ACS/LSM processing other requests
A full panel	A (partially or completely) empty panel
A drive panel	A standard panel
An inside panel	An outside panel

In addition, consider the LSM type for the component you audit, as shown in [Table 13](#). For the 4410, 9310, and L5500 the best time can be achieved when all the cells are full, and worst time when cells are empty.

**Table 13. Average Audit Times for Supported LSMs**

LSM Type	Component	Average Audit Time (Best and Worse)
4410	LSM	3 - 8 hours
9310/L5500	LSM	1.2 - 6 hours

**Table 13. Average Audit Times for Supported LSMs**

LSM Type	Component	Average Audit Time (Best and Worst)
9360	LSM	5 minutes
97xx/L700/180	LSM	1 minute
SL500	LSM	3 minutes
SL8500	LSM	5 minutes

**Note:** These times assume no other active processes. 4410 and 9310 audit times may be reduced if a high percentage of the cells audited are occupied.

## Auditing the SL8500 Library

For specific auditing requirements, refer to [“Please refer to the Readme for PTF830785 for a complete description of the FAIL\\_OVER variable” on page 462](#)

## Auditing the SL500 Library

For specific auditing requirements, refer to [“Auditing the Library” on page 163](#).

## ■ Adding a SCSI or fibre-attached library to ACSLS

Use this procedure when you are adding a SCSI-attached LSM (such as the 9710) to ACSLS. In this procedure, you install the SCSI device driver.

**To add a SCSI-attached LSM to ACSLS, do the following:**

1. Log in as `acsss`.
2. Change to the install directory:  
`cd /ACSSS/install`
3. Log in as `root`.  
`su root`
4. Enter the following command:  
`./install_scsi_sol.sh`
5. Respond to the prompts to install the SCSI device driver.

## ■ Creating an mchanger Device Node

The mchanger is the device driver that communicates between ACSLS and any SCSI library. An mchanger must be created for each SCSI or fibre-attached library that is connected to ACSLS. Adding an mchanger device is part of the installation process. However, there may be times when you need to add a new SCSI library to an existing ACSLS installation. In this case, you must create a new mchanger device node. To do this:

1. Login to the ACSLS server as `root` user and enter your password.

2. Go to the installation directory:

```
cd/export/home/ACSSS/install
```

3. Make sure that the SCSI library is functional and attached to the ACSLS server.

4. Invoke the driver installation script:

```
./install_scsi_sol.sh
```

You are prompted for the specific host-bus adapter and target ID of each library. If you do not know the target ID:

- Access the operator panel on the library
- Select the menu options to display the SCSI target ID

## ■ Removing an mchanger Device Node

You may want to remove an mchanger in the event:

- you have changed the target ID of your library, and want to remove the original node.
- the mchanger device node has been corrupted and you must build a new one

1. Login to the ACSLS server as `root` user and enter your password.

2. Remove the mchanger device(s):

```
rem_drv mchanger
```

The Solaris kernel is rebuilt. All instances of mchanger are removed.

## ■ Using the Extended Store Feature

The following sections provide information about using the Extended Store Feature.

**Note:** This feature does not apply to a single LSM without a pass-thru port.

When a cartridge is mounted, its “home location” is the storage cell from which it was mounted. ACSLS attempts to return a cartridge to its home location after a dismount if that home location is in an LSM that is enabled for the Extended Store feature. Events such as a label mismatch on a dismount, however, may prevent ACSLS from returning a cartridge to its home location. If such an event occurs, the cartridge is stored in the LSM from which it was dismounted.

If you have an Extended Store LSM that you use for cartridge archive, this feature helps to ensure that cartridges mounted from that LSM return to their home locations in that LSM. For example, if Extended Store LSM 3 is enabled for this feature and a cartridge from LSM 3 is mounted on a drive attached to LSM 1, after the cartridge is dismounted, ACSLS will attempt to return the cartridge to its home location in LSM 3. If LSM 3 is not enabled, ACSLS will store the cartridge in a new cell in LSM 1.

You can only enable entire LSMs for this feature. You cannot enable LSM subcomponents, such as a panel or individual cell. To enable an entire ACS for this feature, you must enable each LSM in the ACS.

**Note:** Enabling the Extended Store feature will increase pass-thru activity when cartridges are dismounted. This may significantly decrease library performance.

To enable an LSM for the Extended Store feature, modify the `lsm_fixed_volume` file (found in the `$ACS_HOME/data/external/fixed_volume` directory) shown in [Figure 32 on page 168](#). After you modify the file, stop and restart ACSLS to enable the specified LSMs.

Use the following conventions when you modify the sample file:

- Comment and blank lines are allowed throughout the file.
- Each LSM identifier line consists of the ACS number, followed by a comma, followed by the LSM number. No spaces or tabs are allowed between the ACS number and the LSM number.
- Each LSM identifier must be on a separate line.
- No explicit ordering of the LSM identifier lines is required.
- No explicit end-of-file delimiter is required.

**Figure 32. Sample Extended Store Feature Control File**

```
# This lsm_fixed_volume file must be found in the
# $ACS_HOME/data/external/fixed_volume
# directory. This is a sample lsm_fixed_volume file that may be
# edited your particular configuration.
# Comments may appear anywhere in this file, but must include a
# pound sign in the first column.
# Blanks lines are also allowed throughout the file for
# readability, and
# will be ignored.
# For all the LSM identifiers found in this file, a "best" attempt
# will be
# made to return the volume to its home location at dismount.
# A valid LSM identifier consists of the ACS number, separated by a comma,
# and followed by the LSM number. Leading or trailing blanks are
# ignored.
#   ACS,LSM
#   0,0
#   0,1
# ACS 1, LSM 0 through 3
#   ACS,LSM
#   1,0
#   1,1
#   1,2
#   1,3
```

[Figure 33 on page 169](#) shows an example of a modified control file. In this example, the comment (#) character has been removed from the lines shown in bold to specify that LSMs 0,0 and 0,1 are enabled.

**Figure 33. Modified Extended Store Feature Control File**

```

# This lsm_fixed_volume file must be found in the
# $ACS_HOME/data/external/fixed_volume
# directory. This is a sample lsm_fixed_volume file that may be
# edited your particular configuration.
# Comments may appear anywhere in this file, but must include a
# pound sign in the first column.
# Blanks lines are also allowed throughout the file for
# readability, and
# will be ignored.
# For all the LSM identifiers found in this file, a "best" attempt
# will be
# made to return the volume to its home location at dismount.
# A valid LSM identifier consists of the ACS number, separated by a comma,
# and followed by the LSM number. Leading or trailing blanks are
# ignored.
#   ACS,LSM
#     0,0
#     0,1
# ACS 1, LSM 0 through 3
#   ACS,LSM
#     1,0
#     1,1
#     1,2
#     1,3

```

## ■ Managing a Mixed-Media Library

“Mixed media support” means that ACSLS supports a mixture of transport and media (cartridge) types in the same library. ACSLS mixed-media support prevents the LSM robot from mounting incompatible media types in a transport. For example, in a 9310 LSM, the robot will not mount a DD3A cartridge (helical recording) in a 9490 transport (parallel recording). Note that ACSLS mixed-media support requires media ID labels with media characters on cartridges, as described in “Media Characteristics” in *Product Information*. For more information, see the following sections of this book or the *Product Information*:

- “LSM, Transport, and Media Compatibility” in *Product Information*
- “Transport and Media Compatibility” in *Product Information*
- “Media Characteristics” in *Product Information*
- [“Mixed-Media Restrictions for SCSI-Attached LSMs” on page 171](#)
- [“Setting Scratch Preferences” on page 172](#)

ACSLs does *not*, however, support mixed format recording. Because ACSLS does not have access to the data path to a transport, ACSLS cannot detect and prevent mixed-format recording incompatibilities.

**CAUTION:** The `venter` command does not provide an option to specify the media type of the cartridge you want to enter. In a mixed-media environment, ACSLS cannot prevent transport/media incompatibilities for virtually entered cartridges!

## Displaying the ACSLS Mixed-Media Settings

The `drives_media.sh` utility displays the drive types, media types, and drive-to-media compatibility settings currently supported by ACSLS. As support is added for new drives and media, they are displayed.

To display the ACSLS Mixed-Media settings, enter:

```
drives_media.sh
```

The information is written to the screen (standard output).

To output the ACSLS Mixed-Media settings to files in the `/tmp` directory, enter:

```
drives_media.sh -f
```

The information will be written to three files. (If the files already exist, they will be overwritten.)

```

/tmp/drive_types.txt
/tmp/media_types.txt
/tmp/media_compatibility.txt

```

## Using Mixed-Media Support with 3490E Cartridges

An LSM's vision recognition system can determine the media type if the 3490E cartridge has the "E" media label on the cartridge label edge of the cartridge. The media label is a separate, single character, bar code label below the six-character cartridge label. Cartridges without the media label are considered 3480 cartridges by 1/2" cartridge standards.

StorageTek recommends that when you purchase new 3490E cartridges, you order them with the "E" media label to enable ACSLS to provide automatic mixed media protection for those cartridges. StorageTek strongly suggests that, if practical for your site, you apply "E" media labels to any existing 3490E cartridges without such labels to allow ACSLS to provide automatic protection for those cartridges as well. If you choose to use 3490E cartridges without "E" media labels in a mixed-media environment, you, not ACSLS, must explicitly manage those cartridges. For example:

- A 3490E cartridge without the "E" seventh media character appears as a 3480 cartridge on cartridge reports and as a `query volume` response. You must, therefore, know which 3490E cartridges are incorrectly labelled, and must explicitly mount them in a compatible drive (4490 or 9490).

- Similarly, you must explicitly manage incorrectly labelled 3490E scratch tapes as follows:
  - First, you must create a homogeneous scratch pool containing only the incorrectly labelled 3490E tapes. When you create this pool, do not specify the `overflow` option unless the common scratch pool is a homogeneous pool that contains only 3490E tapes.
  - Next, you must add an entry to the scratch preferences to permit mounting 3480 scratch tapes as an alternate choice for both 4490 and 9490 drives.
  - Finally, when you enter a `mount scratch` command, you must specify the pool ID of the incorrectly labelled 3490E tapes and the `ANY_MEDIA_TYPE` or `ANY_MEDIA_TYPE` keyword.

**CAUTION:** If you modify the scratch preferences file as described, a `mount scratch` command for a 4490 and 9490 drive will mount a 3480 cartridge when the pool no longer contains 3490E cartridges.

- `query mount` and `query mount scratch` commands will not return accurate media type information for incorrectly labelled 3490E cartridges. Again, you must know by volume ID which cartridges that appear as 3480 cartridges are actually 3490E cartridges.

## Mixed-Media Restrictions for SCSI-Attached LSMs

Please note the following restrictions for SCSI-attached LSMs:

- Because some tape drives do not support dynamic write-protect settings by means of the control path, you must use caution when mounting cartridges using the 'read-only' option. All StorageTek drives support this feature. For non-StorageTek drives, you are advised to confirm that dynamic write-protection is supported. Unless the drive supports this feature, you bear the risk of losing data on cartridges that you assumed are to be mounted with the write-disable protection of a 'read-only' mount.
- ACSLS does *not* support the `venter` command for SCSI-attached LSMs with DLT transports.
- You cannot do virtual mounts and dismounts to DLT transports in SCSI-attached LSMs. Compac tape cartridges, therefore, must have external labels to allow ACSLS to manage these cartridges.
- Automatic-cleaning operations in SCSI-attached LSMs is handled by the LSM microcode and not by ACSLS. The LSM control panel provides a menu for users to enable LSM control of automatic-cleaning operations. For more information, see your LSM documentation.
- You can select a normal load or fast load option via the 97xx control panel. Not all tape management systems, however, support the fast load option.

## Setting Scratch Preferences

On scratch mount requests you can explicitly specify the media type you want to use, or you can have ACSLS select a media type.

In order for ACSLS to select a media type, you must pre-define a prioritized list of compatible media types for each drive type. This list is referred to as the “scratch preferences.”

For example, on TimberLine (9490) drives, you may want to use 3490E cartridges before 3490 or 3480 ones. To do this, you would define 3490E cartridges as the highest priority for 9490 drives, followed by 3490 and 3480 cartridges. Note that:

- There is one set of preferences for the entire server; preferences are not defined by client.
- If a compatible media type is not listed for a drive, the media will not be selected.

The following sections describe the user- and system-defined files that ACSLS uses in determining scratch preferences.

### User-Defined Mixed-Media Files

[Table 14](#) describes the user-defined mixed-media files that are located in `$ACS_HOME/data/external/mixed_media/`:

**Table 14. User-Defined Mixed-Media Files**

File	Description
<code>scratch_preferences.dat</code>	User-defined preferences file. Primary source of preference definitions.
<code>scratch_preferences.SAMPLE</code>	Sample preferences file. Can be copied to create the <code>scratch_preferences.dat</code> file.

## System-Defined Mixed-Media Files

[Table 15](#) describes the system-defined mixed-media files that are located in `$ACS_HOME/data/internal/mixed_media/`:

**Hint:** You cannot modify the system-defined mixed-media files.

**Table 15. System-Defined Mixed-Media Files**

File	Description
<code>media_compatibility.dat</code>	System-defined preferences file. Used only if the user-defined file does not exist or is missing a drive type.
<code>drive_types.dat</code>	System-defined list of supported drive types.
<code>media_types.dat</code>	System-defined list of supported media types.

## How ACSLS Uses the Mixed-Media Files

[Table 16](#) describes how ACSLS uses the mixed-media files to select a media type for a scratch mount request.

**Table 16. How ACSLS Uses Mixed-Media Files**

If the	ACSLs does this...
<code>scratch_preferences.dat</code> file ...	
Does not exist.	Uses the definitions in the system file, <code>media_compatibility.dat</code> .
Lists more than one media type for a drive.	Selects the media types in the order listed.
Lists no media type for a particular drive.	Uses the data in the system file, <code>media_compatibility.dat</code> .
Does not list a particular drive type.	Uses the data in the system file, <code>media_compatibility.dat</code> .

## Defining a Scratch Preferences File

Use this procedure to define a `scratch_preferences.dat` file, which contains an ordered list of scratch cartridge types to select for given drive types. ACSLS uses this file for `mount *` command where a media type is not explicitly specified. [Figure 34](#) shows the contents of the `scratch_preferences.SAMPLE` file.

**Figure 34. scratch\_preferences.SAMPLE**

```
Drive Type NameMedia Type Preference Name
4480 3480
4490 3490E
9490 3490E
SD3 DD3A
SD3 DD3B
SD3 DD3C
```

### To define a scratch preferences file, do the following:

1. Log in as `acsss`.
2. Change to the external mixed media directory:
 

```
cd $ACS_HOME/data/external/mixed_media
```
3. Copy the sample scratch preferences file to create the user-defined file:
 

```
cp scratch_preferences.SAMPLE scratch_preferences.dat
```
4. Using a text editor such as `vi`, modify the list of preferences in the `scratch_preferences.dat` file:
  - Follow the instructions in the comments at the top of the file.
  - If you want to use more than one media type for a drive type, enter each media type on a separate line; the order of preference is from top to bottom.
5. Save the file.
6. From a `cmd_proc`, restart ACSLS:
 

```
start
```

See the command, [“set scratch” on page 372](#).

## ■ Managing a Dual-LMU Configuration

ACSL S supports *dual-LMU configurations*, which consist of:

- A *master* LMU that manages an ACS
- A *standby* LMU that automatically switches to master role to manage the ACS if the master LMU fails

Both LMUs are connected to a LAN that is connected to the LSMs. The first LMU powered on is initially the master, while the second LMU powered on is initially the standby. The LMUs periodically check each other's status so the standby can take over the master role if the master fails.

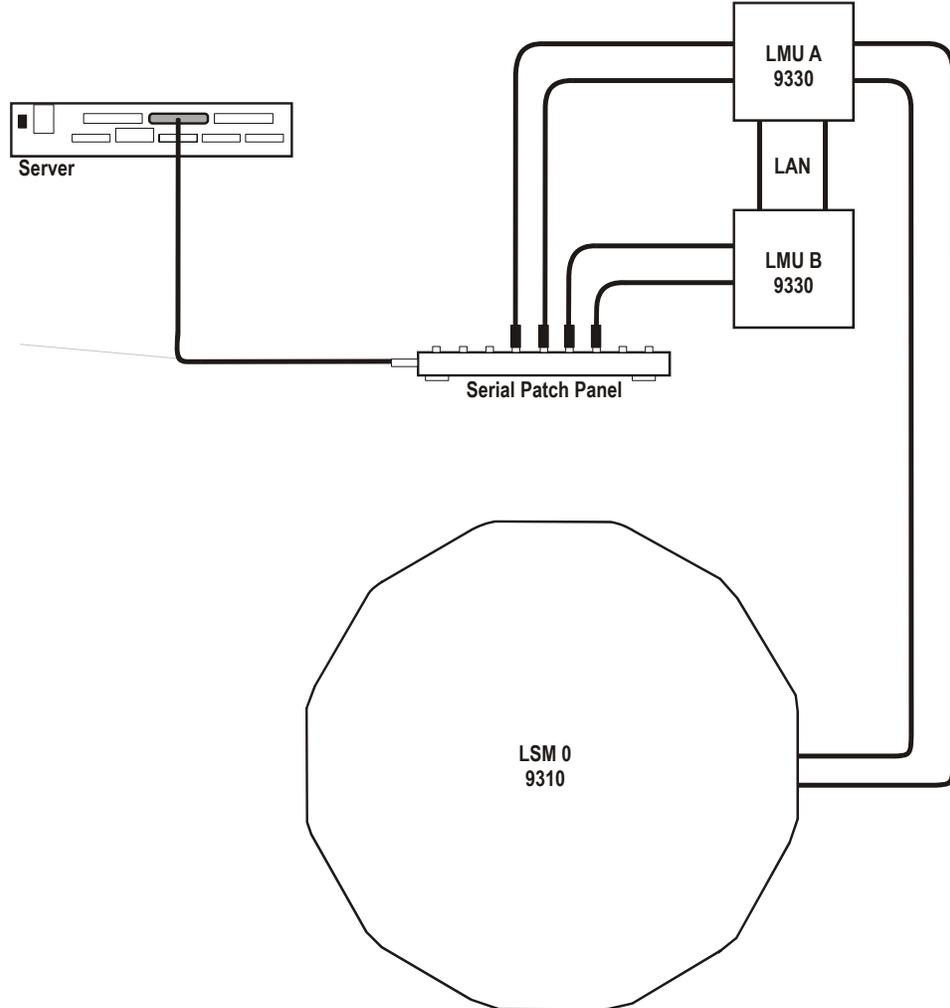
**Note:** ACSLS supports dual-LMU configurations for only the 9330 and L5530 LMUs with host/LMU microcode compatibility Level 12 (or above) loaded. The same microcode level must be loaded in both LMUs. ACSLS communicates with these LMUs by a serial connection, or through TCP/IP. There can be only one Ethernet connection for each TCP/IP connected LMU.

**Limitation:** ACSLS **does not** automatically initiate an LMU switch. Even if communication between ACSLS and the master LMU is lost, ACSLS does not initiate an LMU switch. ACSLS continues trying to communicate with the existing master LMU.

ACSL S dual-LMU support includes:

- ACSLS connects to both master and standby LMUs. ACSLS constantly monitors its connection to both LMUs. When communication is lost, ACSLS reports this condition.
- If the master LMU fails, the standby LMU automatically takes over as the new master LMU. When this happens, ACSLS recognizes the automatic switch and sends requests to the new master LMU. ACSLS also recovers transactions in progress (except for audits).
- You can issue the `switch lmu` command to manually switch ACS management from the master LMU to the standby LMU. ACSLS sends a "Force Switchover to Master" transmission to the standby LMU. The standby takes over as the new master LMU. After a manual switchover, ACSLS recovers outstanding transactions (except for audits).

**Hint:** To further enhance the redundancy of a serial dual-LMU configuration, StorageTek recommends that you use dual serial cables to connect the ACSLS server to *each* LMU. [Figure 35 on page 176](#) shows an example of an ACSLS Sun server in a dual-LMU configuration. In [Figure 35 on page 176](#), a port on the Sun connects to a serial patch panel, which has dual cabling to each LMU.

**Figure 35. ACSLS Server in Dual-LMU Configuration**

S403\_005

ACSLs dual-LMU support also enhances single-LMU configurations because the LMU can IPL without bringing down the ACSLS server.

The `query lmu` command displays LMU and port status for both single-LMU and dual-LMU ACS configurations; for more information see [“query lmu” on page 345](#).

You can also use the `switch lmu` command to manually switch ACS management from the ACS's master LMU to the standby LMU; for more information see [“switch lmu” on page 377](#). You can use manual switchover for hardware maintenance, such as replacing a LAN cable connecting dual LMUs. For example, assume that LMU A is in the master role and LMU B is in the standby role. If you need to replace a LAN cable, you can:

1. Switch to LMU B.
2. Vary the port(s) to LMU A offline.

3. Replace the LAN cable.
4. Vary the port(s) to LMU A back online.
5. Switch back to LMU A.

After an LMU switchover (automatic or manual), ACSLS recovers all outstanding (active and pending) requests. The time to complete each outstanding request during a switchover is increased by the time required to switch between LMUs and recover any preceding outstanding requests. Request recovery, therefore, can take three to five minutes.

## ■ Managing a Dual-LAN Client Configuration

ACSL 5.2 and above supports *dual-LAN client configurations*, which consist of a primary LAN and a secondary (backup) LAN. If the primary LAN fails, the client switches to the secondary LAN. ACSLS removes all outstanding messages to that client and begins communicating via the secondary LAN. Because ACSLS removes all outstanding messages before switching over to the secondary LAN, a client will not receive a success message even though a request completed successfully.

For example, a client requests ACSLS to eject ten cartridges. ACSLS starts ejecting the cartridges, then communications fail between ACSLS and the primary client LAN. ACSLS removes all outstanding messages to that client and begins communicating via the secondary LAN. ACSLS successfully completes ejecting all ten cartridges, but sends no success message to the client. The client must verify that the request completed successfully. In this example, if the client issues a query volume request against the IDs of the ejected volumes, ACSLS returns a `volume not found` error message, which confirms that ACSLS ejected the cartridges.

ACSL 5.2 ensures that any transient requests complete successfully if a LAN communications switchover occurs. After switchover, however, the client must cancel any persistent outstanding requests (such as CAP operations) originally submitted on the primary LAN and resubmit these requests via communications from the secondary LAN. Resources (such as, CAPs, locks, drives, and so forth) allocated via primary LAN communications will remain allocated after switchover to the secondary LAN.

The following sections tell how to configure ACSLS for dual-LAN client operations by:

- Register the IP addresses of primary and secondary LANs by creating a `csc_ip_switch.dat` file
- Install a second ethernet port for a multi-homed ACSLS server
- Set the TCP/IP connection timeout interval to decrease the system switchover time to the backup LAN.

## ■ Registering the IP Addresses of Primary and Secondary LANs

To register the IP addresses of the primary and secondary LANs with ACSLS, create the file `csc_ip_switch.dat` in the `$ACSSS_HOME/data/internal/client_config/` directory. [Figure 36](#) shows an example of a `csc_ip_switch.dat` file.

**Figure 36. Example of `csc_ip_switch.dat` File**

```
#The following entry is System Zed's primary and secondary LAN IP
addresses.
129.80.30.40 129.80.30.50
```

As [Figure 36](#) shows, use a # sign to precede comments. The entries consist of a client system's primary LAN IP address in the left column followed by one or more blanks, then a client system's secondary LAN IP address in the right column. In this example, System Zed's primary LAN IP address is 129.80.30.40 and its secondary LAN IP address is 129.80.30.50.

After you create or update the `csc_ip_switch.dat` file, if ACSLS is running, you must stop and restart ACSLS. If ACSLS is not running, start ACSLS. For more information, see [“Stopping ACSLS” on page 3](#) and [“Starting ACSLS” on page 3](#).

If ACSLS successfully reads the `csc_ip_switch.dat` file on restart, ACSLS logs the success message `2010 I DUAL PATH OPTION ACTIVATED` in the ACSLS event log. Otherwise, dual-LAN support is not activated.

## ■ Installing a Second Ethernet Port for a Multi-Homed ACSLS Server

This section describes the procedure for installing a second ethernet port for a multi-homed ACSLS server. The second ethernet port provides the attachment to the second control path adapter, which controls the backup LAN. This procedure requires the Sun SBUS Buffered Ethernet card. You can order this Ethernet card from Sun as part number X1053A or from StorageTek as part number 309479501.

The installation procedure includes:

- Installing the hardware and rebuilding the kernel
- Defining a host name for the new ethernet port
- Creating the `/etc/notrouter` file

To install a second ethernet port, do the following:

1. Install the Sun SBus Buffered Ethernet card according to the manufacturer's instructions.
2. Reconfigure the system kernel for the new device:
  - a. Power the system on. When it begins to boot, press `[[STOP]]-[[A]]` to enter the PROM monitor.
  - b. At the ok prompt, boot the server:

```
boot -r
```

3. When the boot has completed, login as root.
4. Create a hostname for the second ethernet port.

```
echo 2nd_host_name > /etc/hostname/hme1
```

Where *2nd\_host\_name* is the hostname for the second ethernet port.

5. Enter the following:

```
touch /etc/notrouter
```

This creates the `/etc/notrouter` file.

6. Reboot the server:

```
reboot
```

This completes this procedure.

## ■ Setting the TCP/IP Connection Timeout Interval

The UNIX system variable `tcp_ip_abort_cinterval` sets the TCP/IP connection timeout interval between the client and the ACSLS server. Changing the default value (180 seconds) of this variable may decrease the switchover time to the backup LAN. Note, however, that actual switchover time depends on the configuration and type of failure. For example, if an ACSLS server port fails and multiple clients are communicating via this port, ACSLS serially recovers communications with each client. Multi-client recovery for a failing port therefore takes longer than recovery with a single client communicating with a failing port.

To decrease the switchover time to the backup LAN, do the following:

1. Log in as `root` on the ACSLS server.
2. From the prompt, enter the following:

```
/usr/sbin/ndd -set /dev/tpc tcp_ip_abort_cinterval 15000
```

This command changes the TCP/IP connection timeout interval to 15 seconds (the default is 180 seconds).

**Hint:** For Solaris only: To make this command persistent (across server reboots), add the command to the `/etc/rc2.d/S69inet` file in the “Set configurable parameters” section.

## ■ Registering for Event Notification

Event notification allows tracking of events occurring in tape libraries for ACSAPI clients. This feature is provided by the CSC Toolkit. Specifically, with event notification, the ACSAPI client can do the following:

- Register for library resource events and/or volume events
- Unregister for these events
- Check registration status for library events and be notified when they occur

Client registration requests and event notification messages are delivered to the client until the registration is dropped. Event notification periodically checks the registration status of the client to verify that the client is alive. This avoids sending responses to clients that are no longer active and avoids the unnecessary use of network resources.

The following types of events can be tracked:

- Addition and deletion of volumes and when maximum usage is exceeded for cleaning cartridges in the ACSLS database.
- Changes in library component status such as an LSM or drive changing from online to offline, diagnostic, or recovery; or a CAP that is opened or closed.
- Hardware failures such as an inoperable robotic hand.

---

Cartridge management consists of the following:

- Manual Cartridge Delete Utility
- Absent and ejected cartridge support
- Cartridge recovery
- Missing cartridges
- CAP usage
- Entering cartridges
- Ejecting cartridges
- Managing scratch cartridges
- Cleaning cartridges

## ■ Using the Manual Volume Delete Utility

The Manual Volume Delete utility, `del_vol`, allows you to access a volume that is in an offline, unavailable LSM. If you manually remove the cartridge from the LSM and try to re-enter it into another LSM, ACSLS will issue a `duplicate volume` message and will not enter the cartridge. Using the `del_vol` utility, you can first delete the volume from the database, then manually remove it from the offline LSM and successfully re-enter it into an online LSM.

The Manual Volume Delete (`del_vol`) utility now retains volumes as absent, with the option to delete the volume. The volume can be deleted without waiting for the expiration of an absent or ejected status.

### Notes:

- To remove a cartridge from an online LSM, issue an `eject` command for the cartridge.
- ACSLS and the database must be up and running to use this utility. Do not run `del_vol` while the system is in recovery; unpredictable results may occur.

For more information about this utility, see [“del\\_vol” on page 254](#).

### Note:

To delete a cartridge using the `del_vol` utility:

1. Log in as `acsss`.
2. Delete the cartridge:

```
del_vol vol_id
```

For more information see about switch options you can use with `del_vol`, see [“del\\_vol” on page 254](#).

## ■ Using Absent and Ejected Cartridge Support

Absent cartridge support in ACSLS marks cartridges that cannot be found in the library as absent, instead of deleting them. If these cartridges are later found in the library, ACSLS changes them to active status instead of re-adding them to the database. Reactivation preserves settings, such as pool, volume access control ownership, and locks.

Similarly, ejected cartridge support retains cartridge information when cartridges are ejected. The cartridges are reactivated when they are re-entered.

Absent and ejected volume support is enabled when the `ABSENT_VOLUME_RETENTION_PERIOD` is set to a non-zero number of days. The default value is 5 days.

Additional aspects of absent and ejected cartridge support include:

- The Manual Volume Delete (`del_vol`) utility will retain volumes as absent unless the `-d` option is specified. If this option is specified, the volume is deleted without waiting for the expiration of an absent or ejected status.
- ACSLS improves volume recovery by searching for volumes that are not found in their expected locations in the library. ACSLS searches all recorded locations instead of automatically deleting the volume.
- Clients can specify, through the `ENABLE_STATUS_VOLUME_ABSENT` and `ENABLE_STATUS_VOLUME_MISSING` configuration settings, whether they want absent, ejected, and missing statuses to be reported through the ACSAPI.
- The `volrpt` utility with the `-i` option will report volume records with a status of absent or ejected. By default, `volrpt` does not report absent or ejected volumes.

## ■ Absent, Ejected and Missing Cartridges

ACSLs reports three cartridge (volume) statuses:

- missing

The cartridge cannot be located in the library, and at least one recorded location for the cartridge cannot be searched because the LSM is offline or a drive is not communicating. The information about the cartridge has been retained.

- absent

The cartridge cannot be located in the library. All recorded locations for the cartridge have been searched, and the cartridge is not in any of them. The information about the cartridge is retained. If the cartridge is found or re-entered into the library (before the retention period expires), it is reactivated.

- ejected

The cartridge was ejected. The information about the cartridge is retained, and if the cartridge is found or re-entered (before the retention period expires), it is reactivated.

## Cartridge (Volume) Status Reporting

ACSLs reports cartridges (volumes) with the status “missing”, “absent,” or “ejected,” differently in response to ACSLS commands than it does in response to ACSAPI requests.

The information displayed in response to ACSLS commands identifies a cartridge as “missing”, “absent”, or “ejected.”

However, the cartridge status information ACSLS displays in response to ACSAPI requests is governed by the following ACSLS dynamic variables:

1. missing
  - a. If the ACSLS dynamic variable *ENABLE\_STATUS\_VOLUME\_MISSING* is TRUE, ACSLS reports: *STATUS\_VOLUME\_MISSING*.
  - b. If the ACSLS dynamic variable *ENABLE\_STATUS\_VOLUME\_MISSING* is FALSE, ACSLS reports: *STATUS\_VOLUME\_IN\_TRANSIT*.

2. absent
  - a. If the ACSLS dynamic variable *ENABLE\_STATUS\_VOLUME\_ABSENT* is TRUE, ACSLS reports: *STATUS\_VOLUME\_ABSENT*
  - b. If the ACSLS dynamic variable *ENABLE\_STATUS\_VOLUME\_ABSENT* is FALSE, ACSLS treats the volume as if it had been deleted from the ACSLS database and reports: *STATUS\_VOLUME\_NOT\_IN\_LIBRARY*.
3. ejected
  - a. If the ACSLS dynamic variable *ENABLE\_STATUS\_VOLUME\_EJECTED* is TRUE, ACSLS reports: *STATUS\_VOLUME\_EJECTED*
  - b. If the ACSLS dynamic variable *ENABLE\_STATUS\_VOLUME\_EJECTED* is FALSE, ACSLS treats the volume as if it had been deleted from the ACSLS database and reports: *STATUS\_VOLUME\_NOT\_IN\_LIBRARY*.

#### ABSENT\_VOLUME\_RETENTION\_PERIOD Dynamic Variable

The *ABSENT\_VOLUME\_RETENTION\_PERIOD* dynamic variable controls how long absent and ejected volumes are retained in the ACSLS database and specifies the number of days these volumes are retained. There are two special values:

- The value 0 (zero) days specifies that volumes are deleted and will not be marked absent or ejected. (This is the behavior of ACSLS releases previous to ACSLS 6.1.)
- The value 999 days specifies that absent and ejected volumes are retained forever in the database.

## ■ Cartridge Recovery

Cartridge Recovery (acscr) is an ACSLS internal process that is called to resolve discrepancies whenever the actual content of a storage cell or tape drive does not match the information saved in the ACSLS database. It does this by having the library examine and drives, and then updating the ACSLS database with the results. If Cartridge Recovery finds a discrepancy (e.g., a cartridge that is recorded at another location), it creates another recovery request and adds it to its request queue. (This is called a “cascade”.)

Other processes pass recovery requests to Cartridge Recovery when they encounter a discrepancy between the ACSLS database and the actual contents of the library. Because of this, Cartridge Recovery is the central location where cartridges are marked as missing, changed to absent, and reactivated. Thus, what appears to be the behavior of many other ACSLS

commands and utilities is actually done by Cartridge Recovery when it updates the database to match the information reported by the library.

When other processes pass recovery requests to Cartridge Recovery, they can either:

1. continue and let Cartridge Recovery continue asynchronously (Cartridge Recovery proceeds independently), or
2. if they need a specific cartridge that ACSLS can't locate, wait for Cartridge Recovery to finish processing this recovery request and report what it found.

## ■ Missing Cartridges

A cartridge is marked missing when:

- Cartridge Recovery cannot find a cartridge in the library, and
- it cannot examine all recorded locations for a cartridge (home cell and drive, if the cartridge has a recorded drive location).

For example, when Cartridge Recovery cannot examine the home cell in an offline LSM or an offline drive, and when it doesn't find the cartridge in other locations, it marks the cartridge missing.

Cartridge Recovery preserves the cartridge's home location unless it examines the cartridge's home cell and finds another cartridge there. In this situation it marks the cartridge “homeless,” with a minus one (-1) in the `home_lsm` field.

When Cartridge Recovery finds a cartridge that was missing, it changes that cartridge's status to “home” or “in drive” in the database, depending where it found the missing cartridge.

1. If the cartridge is found in a cell other than its recorded home cell, Cartridge Recovery checks the cartridge's home cell to see whether it has found a duplicate cartridge.
2. If the cartridge is not in its recorded home cell, Cartridge Recovery records the cell in which it was found as its new home cell.
3. If the new cartridge is a duplicate, Cartridge Recovery reports this on the Event Log. The duplicate cartridge is **not** ejected.
4. If Cartridge Recovery finds a “homeless” cartridge in a drive, it does not assign a new home cell. When the cartridge is dismantled, the dismant process assigns a new home cell.

## ■ Absent and Ejected Cartridges

### Cartridges Not Found

When Cartridge Recovery can examine all recorded locations and it cannot find a cartridge:

1. If the *ABSENT\_CARTRIDGE\_RETENTION\_PERIOD* is 0, Cartridge Recovery
  - deletes the cartridge record from the database.
  - marks the cell record in the database for the cell that was the cartridge's home cell as "empty."
2. If the *ABSENT\_CARTRIDGE\_RETENTION\_PERIOD* is greater than 0, Cartridge Recovery
  - changes the status of the cartridge record in the database to "absent" if the cartridge has not already been marked absent or ejected.
  - records the cartridge as "homeless" (with a minus one (-1) in the *home\_lsm* field).
  - marks the cell record in the database of the cartridge's former home cell as "empty."

### Cartridges Found

If Cartridge Recovery finds an ejected or absent cartridge, it reactivates the cartridge.

If the ejected or absent cartridge is found in a storage cell, this becomes its new home cell, and Cartridge Recovery changes the cartridge's status to "home" in the database.

If the cartridge is found in a drive, ACSLS assigns a new home cell when the cartridge is dismounted.

## ■ Populating the LSM

An essential requirement for the library and ACSLS to function properly is the availability of a few free cells in each LSM to accommodate dismount, pass through, and eject operations. You should reserve at least one free cell for each tape drive installed in each LSM.

To determine the free cell count of an LSM, issue the command:

```
query lsm lsm_id
```

**Note:** In the SL8500, each rail is defined as an LSM.

## ■ Using the CAP

### CAP Types

Each type of CAP has a standard capacity and method for loading it with cartridges. An LSM may have more than one type of CAP. [Table 17](#) shows the supported CAP types, identifiers and capacities, and loading methods.

**Table 17. CAP Types**

CAP Type	Identifier & Capacity	Loading Method
Standard	CAP00; holds 21 cartridges.	Cartridges are loaded directly into the CAP cells.
Enhanced (4410 and 9310)	CAP00 and CAP01; each holds 40 cartridges.	Cartridges are placed into removable magazines which are loaded into the CAP.
9360	CAP00 holds 20 cartridges; optional CAP01 holds 30 cartridges.	Cartridges are placed into removable magazines which are loaded into the CAP.
Priority (PCAP)	CAP02; holds one cartridge.	Cartridges are entered one at a time, directly into the CAP.
9710 or 9740 CAP	CAP00, holds 14 cartridges or magazine, which holds 10 cartridges.	Cartridges are loaded directly into the CAP cells or placed into the removable magazine which is loaded into the CAP.
9714, 9730, or 9738 CAP	CAP00, holds one cartridge	Cartridges are loaded directly into the single-cell CAP.
L180	CAP00, holds 10 cartridges.	5 cartridges are placed into each of 2 removable magazines which are loaded into the CAP.
L700	CAP00 and optional CAP01; each holds 20 cartridges.	5 cartridges are placed into each of 4 removable magazines which are loaded into the CAP.

**Table 17. CAP Types**

CAP Type	Identifier & Capacity	Loading Method
SL500	CAP00; holds between 5 and 25 cartridges	5 cartridges are placed into removable magazines which are loaded into the CAP. One magazine in base module; 2 magazines in expansion modules that contains a CAP.
SL8500	CAP00 and optional CAP01; each holds 39 cartridges.	13 cartridges are placed into each of 3 removable magazines which are loaded into the CAP.

## CAP States

A CAP's state determines whether it is available for entering and ejecting cartridges. [Table 18](#) describes the valid CAP states. See [“Displaying CAP Information” on page 191](#) for procedures for determining the CAP state. See the command, [“vary” on page 382](#) for information about changing device states.

**Note:** Refer to [“SL8500 CAP Behavior” on page 453](#) for specifics regarding the LS8500 library. Refer to [“SL500 CAP Behavior” on page 500](#) for specifics regarding the SL500 library.

**Table 18. CAP States**

State	Description	How requests are handled
diagnostic	The CAP is available for diagnostic activity without interference from client applications.	<ul style="list-style-type: none"> <li>Requests from client applications are rejected.</li> <li>Requests from the cmd_proc are processed.</li> </ul>
offline	The CAP is logically disabled.	<ul style="list-style-type: none"> <li>All requests are rejected.</li> </ul>

**Table 18. CAP States**

<b>State</b>	<b>Description</b>	<b>How requests are handled</b>
offline-pending	Transition state. Occurs when the CAP is taken from online to offline.	<ul style="list-style-type: none"> <li>• All new requests are rejected.</li> <li>• Current and pending requests are processed to completion.</li> </ul>
online	Normal operating state.	<ul style="list-style-type: none"> <li>• All requests are accepted and processed.</li> </ul>
recovery	Transition state. Occurs when the CAP is taken from offline to online.	<ul style="list-style-type: none"> <li>• New requests are rejected.</li> </ul>

## CAP Modes

The CAP mode controls how a CAP will be used for cartridge enters and ejects. [Table 19](#) describes the valid CAP modes. “[Displaying CAP Information](#)” [on page 191](#) for procedures for determining the CAP mode. See the command, “[set cap mode](#)” [on page 365](#) for information about changing the CAP mode.

**Hint:** You cannot change a CAP's mode while the CAP is in use. That is, if the door is open during either manual or automatic enter operations, you cannot change its mode until you complete the enter operation.

Table 19. CAP Modes

Mode	Description	Effects on enter/eject
automatic	<p>The CAP is unlocked when not in use. This is the initial mode for all priority CAPs.</p> <p><b>Note:</b> You cannot set the CAP mode to automatic in a partitioned library.</p>	<ul style="list-style-type: none"> <li>You can enter cartridges without explicitly issuing an <code>enter</code> command. The enter is initiated when you open the CAP door, place the cartridge(s) inside, and close the CAP.</li> </ul> <p><b>Note:</b> You cannot cancel an automatic enter operation that is in progress using the <code>cancel</code> command. To terminate an automatic enter in progress:</p> <p><i>If the CAP door is open, remove all the cartridges and close the door.</i></p> <p><i>If the CAP door is closed and the cartridges are being moved into the library, you must allow the remaining cartridges to be entered into the library. The enter then terminates.</i></p> <ul style="list-style-type: none"> <li>To eject cartridges you must explicitly issue an <code>eject</code> command. You can either specify the <code>cap_id</code> on the command or allow ACSLS to automatically select a CAP, based on previously defined CAP priorities.</li> </ul>
manual	<p>The CAP is locked when not in use. This is the initial mode for all multi-cartridge CAPs.</p>	<ul style="list-style-type: none"> <li>You can enter or eject cartridges only after explicitly issuing a command. You either specify the <code>cap_id</code> on the command, or allow ACSLS to automatically select a CAP, based on previously defined CAP priorities.</li> </ul>
<p><b>Note:</b> Some client applications require CAPs to be in manual mode. See the documentation for your tape management system.</p>		

## CAP Priorities

CAP priorities specify how ACSLS automatically selects CAPs when the CAP request specifies an asterisk (\*) for the CAP ID. [Table 20](#) describes the CAP priorities and their effect. See [“Displaying CAP Information” on page 191](#) for procedures for determining the CAP priority. See [“set cap priority” on page 367](#) for information about changing the CAP priority.

**Table 20. Cap Priorities**

Priority	Effect
16 (highest)	first used
15 (next highest)	next used
-	
-	
-	
1 (lowest)	last used
0	never automatically selected (initial priority for all CAPs)

CAP priorities and automatic CAP selection apply to the following commands:

- `audit`
- `eject`
- `enter`
- `venter`

When you enter any of these commands with an asterisk (\*) as the `cap_id`, ACSLS automatically selects an available CAP with highest non-zero priority for each ACS specified in the request.

## Displaying CAP Information

Following are some guidelines for using the `query cap` command to display current CAP information.

To display information for selected CAPS, enter:

```
query cap cap_id cap_id ...
```

To display information for all CAPS in the library, enter:

```
query cap all
```

Figure 37 shows a typical example of the results of a query cap command.

**Figure 37. Example of Query CAP command**

```
ACSSA> query cap 0,0,0,0,0,1,0,1,0
2004-06-30 13:15:46          CAP Status
Identifier priority Size  State  Mode      Status
000          1       21  online automatic enter
001          0        4  online  manual  eject
010          7       21  offline automatic available
```

## CAPs Must be Empty Before Enters and Ejects Terminate

ACSLs requires that a CAP be empty before it is available for enters and ejects. During initialization, or when an ACS or LSM comes online, ACSLS requests that you remove any cartridges found in a CAP. All cartridges must be removed from the CAP before ejects terminate.

During an enter, if cartridges with missing or invalid external labels are present, they are not moved into the library. The CAP is unlocked, and you must remove the invalid cartridges before the enter can proceed.

**Note:** With a 9310 and other libraries, the robot stops scanning when it encounters an empty CAP cell. A CAP with only the first cell empty appears to be completely empty.

## ■ Entering Cartridges

You can choose to enter cartridges manually or automatically.

- To enter cartridges manually, you need to issue the `enter` command. This unlocks the CAP so cartridges can be entered.
- An automatic enter is initiated by opening a CAP that is in automatic mode. When a CAP is in automatic mode, you need not issue an enter command.

The following steps describes the enter process:

1. Once you start the enter, the CAP is unlocked and reserved. It cannot be used by another host.
2. After you open the CAP, place the cartridges into the CAP, and close the CAP, the CAP is locked.

The ACSLS library robot inspects/audits the cartridges in the CAP. All cartridges being entered must have valid external labels that do not duplicate other `vol_ids` already managed by this ACSLS server.

Note: Virtual enter allows you to enter unlabeled cartridges into some libraries.

3. ACSLS allocates home cells in the library to valid cartridges and moves them to their assigned home cell locations.

Duplicate cartridges and cartridges without external labels are left in the CAP and must be removed.

4. Upon completion, the CAP is unlocked so more cartridges can be entered.
  - If the CAP is in automatic mode, the automatic enter is complete and the CAP is unreserved and available.
  - If this is a manual enter, the CAP is still reserved for the manual enter. To terminate the manual enter, cancel it with either the cancel command or with <CTRL>+C at the `cmd_proc` where the enter was started.

For additional information on the enter command, refer to [“enter” on page 318](#).

**Note:** If cartridge tracing is enabled, the event log records all cartridge enters.

**Table 21. Cartridge Enter Commands**

Task	Command
Entering Cartridges in Automatic Mode	<code>set cap mode automatic cap_id</code>
Entering Cartridges in Manual Mode	<code>enter cap_id</code>
Entering Cartridges with Virtual Labels (venter)	<code>venter cap_id vol_id vol_id</code>  Do <i>not</i> open the LSM door and place cartridges with missing or unreadable labels in a storage cell because ACSLS cannot manage these cartridges. During an audit, ACSLS will eject cartridges with missing or unreadable labels that were ventered.

## Terminating an Enter Request

Use these procedures to terminate or cancel a current or pending manual enter or virtual enter.

**Note:** You cannot cancel an automatic enter operation that is in progress using the cancel command. To terminate an automatic enter in progress:

- If the CAP door is open, remove all the cartridges and close the door.
- If the CAP door is closed and the cartridges are being moved into the library, you must allow the remaining cartridges to be entered into the library. The enter then terminates.

To cancel a manual enter

1. Display all current and pending library activity:

```
query request all
```

2. Note the *request\_id* of the enter/venter request you want to cancel.

3. From the *cmd\_proc*, enter:

```
cancel request_id
```

where *request\_id* is the identifier of the request you want to cancel.

4. Wait for the CAP to unlock, open the CAP, and remove all cartridges.

The *cmd\_proc* will display a message indicating the number of cartridges entered into the library before the cancel request was received. These cartridges will remain under ACSLS control.

See also:

- [“query request” on page 358](#)
- [“cancel” on page 300](#)

## ■ Ejecting Cartridges

To eject cartridges from the library you need to issue the `eject` command.

The following steps describes the eject process:

1. Once you start the eject, the CAP is locked. It cannot be used by another host.
2. The robot places the specified cartridge(s) in the designated CAP, then ACSLS makes the cell locations where the cartridges were stored available for other cartridges.

3. Open the CAP, remove all cartridges from the CAP, and close the CAP. ACSLS then inspects the CAP to ensure it is empty. The CAP is now available for another operation, such as enter or audit.

Note: If you specify more than a CAP full of cartridges on the eject command, empty the CAP when it fills, close the CAP, and ACSLS continues the eject process until all cartridges are ejected.

For additional information on the eject command, refer to [“eject” on page 314](#).

**Note:** If cartridge tracing is enabled, the event log records all cartridge ejections.

## ■ Managing Scratch Cartridges

Scratch cartridges either contain no data or data that can be overwritten. A user or application mounts a scratch cartridge to write new data on that cartridge.

You must ensure that the library has enough available scratch cartridges to satisfy scratch mount requests. For more information see:

- [“Adding Scratch Cartridges to the Library” on page 196](#)
- [“Rebalancing Scratch Pools” on page 196](#)

The following sections provide additional information about managing scratch cartridges and scratch pools:

- [“Displaying Scratch Pool and Scratch Cartridge Information” on page 195](#)
- [“Deleting Scratch Pools” on page 197](#)
- [“Mounting Scratch Cartridges” on page 197](#)
- [“Unscratching Cartridges” on page 198](#)

## Displaying Scratch Pool and Scratch Cartridge Information

To display scratch pool information, use the following ACSLS functions:

`query pool`

Displays scratch pool attributes, see [“query pool” on page 355](#).

`query scratch`

Displays scratch cartridge information, see [“query scratch” on page 359](#).

`query mount *`

Displays the status of media-compatible cartridges for a specified scratch pool (and, optionally, for a specific cartridge media type within the pool), see [“query mount \\*” on page 353](#).

customized cartridge report

Can be created to report selected scratch cartridge information, see [“Creating a Custom Volume Report” on page 233](#).

## Adding Scratch Cartridges to the Library

Use this procedure to add scratch cartridges to the library.

To add scratch cartridges to the library:

1. If necessary, create a new scratch pool:  
For more information, see the [“define pool” on page 306](#)
2. Enter scratch cartridges into the library.  
For more information, see [“Entering Cartridges” on page 192](#)
3. Define the cartridges you entered in Step 2 as scratch cartridges and assign them to a scratch pool.  
For more information, see [“set scratch” on page 372](#)

## Rebalancing Scratch Pools

Use this procedure to rebalance scratch pools by moving scratch cartridges from one pool to another.

To rebalance scratch pools:

1. To display the attributes of all scratch pools, enter:  

```
query pool all
```

  
For more information see [“query pool” on page 355](#)
2. Use the query scratch command to display the IDs of scratch cartridges in pools you want to rebalance.  
For more information, see [“query scratch” on page 359](#)
3. Use the set scratch command to move scratch cartridges from one pool to another.  
For example, to move cartridges YUMA20 through YUMA80 (which currently reside in pool 5) to pool 10, enter:  

```
set scratch 10 YUMA20-YUMA80
```

  
For more information, see [“set scratch” on page 372](#)

## Deleting Scratch Pools

To manage scratch pools, you may want to delete any scratch pools that no longer contain scratch cartridges. You cannot delete the common pool (Pool 0). Note that you can delete only empty scratch pools; you cannot delete a scratch pool if it contains either data or scratch cartridges. You can, however, use [“Deleting All Empty Pools” on page 197](#) to delete all empty pools (ACSLs will not delete any pools that contain scratch or data cartridges).

### Emptying a Scratch Pool

Use this procedure to empty a scratch pool before deleting it.

To empty a scratch pool:

1. To move data cartridges out of the pool, enter:

```
set scratch off 0 vol_id volrange ...
```

Where the *vol\_id* or *volranges* specify the data cartridges you want to move to the common pool (pool 0). For more information, see [“set scratch” on page 372](#).

2. To move scratch cartridges out of the pool, do one of the following:
  - Move the cartridges to another pool.
  - See [“Ejecting Cartridges” on page 194](#). If you eject scratch cartridges, however, ACSLS no longer manages these cartridges. If you later want to use these cartridges, you must reenter them and assign them to a scratch pool.

### Deleting a Single Pool

To delete a single pool:

```
delete pool pool_id
```

### Deleting All Empty Pools

The `delete pool all` command deletes only empty scratch pools, not pools that contain scratch or data cartridges.

To delete all empty pools:

```
delete pool all
```

## Mounting Scratch Cartridges

Use the following procedures to mount scratch cartridges in single-media and mixed-media environments.

## Single-Media Environments

To mount a cartridge from a specified pool:

```
mount * drive_id pool_id
```

If no cartridge is available from the specified pool and the pool has been set for “overflow,” ACSLS will select a cartridge from the common pool (pool 0).

To mount a cartridge from the common pool:

```
mount * drive_id
```

## Mixed-Media Environments

To mount a scratch cartridge with a specified media type from a specified pool:

```
mount * drive_id pool_id media media_type
```

If no cartridge is available from the specified pool and the pool has been set for `overflow` ACSLS will select a cartridge with the specified media type from the common pool (pool 0).

To mount a scratch cartridge from a specified pool with a media type determined by scratch preferences:

```
mount * drive_id pool_id media *
```

If no cartridge is available from the specified pool and the pool has been set for `overflow` ACSLS will select a cartridge from the common pool (pool 0) according to the defined scratch preferences.

To mount a cartridge from the common pool with a specified media type:

```
mount * drive_id media media_type
```

To mount a cartridge from the common pool with a media type determined by scratch preferences:

```
mount * drive_id media *
```

See also:

- [“mount \\*” on page 330](#)
- [“query commands” on page 337](#)
- [“Setting Scratch Preferences” on page 172](#)

## Unscratching Cartridges

Use this procedure to “unscratch” cartridges (return them to data cartridge status) that were scratched in error.

To unscratch cartridges:

1. Use the `query pool` and `query scratch` commands to display the cartridge and pool IDs of the cartridges to unscratch.

For more information see [“query pool” on page 355](#) and [“query scratch” on page 359](#).

2. To unscratch the selected cartridges, enter:

```
set scratch off 0 vol_id volrange ...
```

Where the `vol_id` or `volranges` specify the cartridges you want to change from scratch mode and move them to the common pool (pool 0). For more information, see [“set scratch” on page 372](#).

## ■ Cleaning Cartridges

ACSL S controls automatic cleaning for HLI-attached libraries (SL8500, L5500, 9300, 9740, and 4400 serial or TCP/IP attached libraries), but not for SCSI-attached libraries.

Cartridges must be cleaned periodically to prevent damage to them and the tape media. Transport control units track how much tape passes through each transport and send a message to ACSLS when a transport requires cleaning.

### Automatic Cleaning

If auto-cleaning is enabled, ACSLS automatically mounts a cleaning cartridge on the transport when required. If all cleaning cartridges are expired (the `max_usage` value is exceeded), ACSLS honors the original mount request without cleaning the transport. For that mount and for each subsequent mount to the uncleaned drive, ACSLS posts message 376N to the event log. Add more cleaning cartridges of compatible media with the drive type as described in [“Defining Cleaning Cartridges” on page 200](#).

If auto-cleaning is disabled, ACSLS logs a message in the event log and displays cleaning messages at the `cmd_proc` when the transport requires cleaning. You then must manually mount a cleaning cartridge.

Use the `acsss_config` configuration program to enable or disable auto-cleaning. In addition, with `acsss_config` you can specify how cleaning cartridges are ordered for selections and queries. For more information, see [Chapter 7, “ACSL S Configuration.”](#)

**Note:** You cannot enable auto-cleaning on SCSI-attached libraries.

For more information on cleaning cartridges, refer to:

- [“Defining Cleaning Cartridges” on page 200](#)
- [“Ejecting Used Cleaning Cartridges” on page 201](#)

- [“Manually Cleaning a Transport” on page 201](#)
- [“Cleaning Cartridges on SCSI-Attached LSMs” on page 202](#)
- [“Correcting Cleaning Cartridge Problems” on page 202](#)

## Defining Cleaning Cartridges

You must use the `set clean` command to define a cleaning cartridge because ACSLS does not define cleaning cartridges by `vol_id`.

When you define cleaning cartridges, make sure to:

- Use cleaning cartridges whose media types are compatible with each transport type in your library. ACSLS will automatically select the correct type of cartridge for each cleaning operation.
- Define at least one cleaning cartridge for each transport in your library.

To define cleaning cartridges to ACSLS:

1. Make the CAP ready for entry.

See [“Entering Cartridges” on page 192](#) for more information.

2. Enter the cleaning cartridges.

The `cmd_proc` displays messages with the cartridge IDs of the cartridges you enter.

3. To set the cleaning cartridge attribute:

```
set clean max_usage vol_id | volrange
```

Where:

*max\_usage* is the number of times a cleaning cartridge is used before ACSLS stops selecting the cartridge to clean cartridges.

*vol\_id | volrange* specifies the cleaning cartridge or range of cartridges that you entered in Step 2.

See also:

- [“enter” on page 318](#)
- [“set clean” on page 369](#)
- [“query clean” on page 342](#)
- [“volrpt” on page 281](#)

## Ejecting Used Cleaning Cartridges

ACSLs logs a message to the event log when a cleaning cartridge has reached the maximum use specified on the set clean command. ACSLS leaves the cartridge in the library, but will no longer select it for cleaning. Eject the used cleaning cartridge and enter a replacement.

To eject used cleaning cartridges:

1. To eject the cleaning cartridges, enter:

```
eject cap_id vol_id|volrange
```

Where:

*cap\_id* specifies the CAP used to eject the cleaning cartridges.

*vol\_id|volrange* specifies the IDs of the cleaning cartridges to eject.

2. Remove the cleaning cartridges.

See also:

- [“query clean” on page 342](#)
- [“eject” on page 314](#)
- [“volrpt” on page 281](#)

## Manually Cleaning a Transport

Use this procedure to clean a transport when auto-clean is disabled or is not working.

To manually clean a transport:

1. Determine which cleaning cartridge types are compatible with the drive to be cleaned.

Refer to the *Product Information Guide* and look in the Transport and Media Compatibility table for a list of the cleaning cartridges for each drive type

If you do not have access to this manual, you can download it from the Customer Resource Center (CRC).

2. Display the available cleaning cartridges:

```
query clean all
```

To display all compatible cleaning cartridges in the same ACS as the drive, use the display command.

```
display volume * -home acs,*,*,*,* -media media_type
```

To display the cartridge's maximum cleaning usage and current usage:

```
display volume * -home acs,*,*,*,* -media media_type
-field vol_id acs lsm media max_use access_count
```

3. Select a compatible cleaning cartridge from those listed and mount it on the transport:

```
mount vol_id drive_id
```

4. After the transport is cleaned and the cleaning cartridge is unloaded, dismount the cleaning cartridge:

```
dismount vol_id drive_id
```

See also:

- [“query clean” on page 342](#)
- [“display volume” on page 409](#)
- [“mount” on page 327](#)
- [“dismount” on page 310](#)

## Cleaning Cartridges on SCSI-Attached LSMs

You cannot use the `acsss_config` configuration program to enable auto-cleaning for drives attached to SCSI-attached LSMs. Using ACSLS, you can only clean these drives by manually mounting a cleaning cartridge. The LSM hardware, however, lets you enable auto-cleaning via the LSM control panel. For more information, see your LSM documentation. For 9714, 9730, or 9738 LSMs, if you either enable or disable auto-cleaning via the control panel, for the change to take effect you must do the following:

- Stop ACSLS; see [“Stopping ACSLS” on page 3](#).
- Rerun `acsss_config` without changing any options but ensuring that you enter Y to the `Build/verify library configuration?` prompt.
- Restart ACSLS; see [“Restarting ACSLS” on page 12](#).

## Correcting Cleaning Cartridge Problems

The following procedures tell how to:

- Change a cleaning cartridge's maximum use count. For example, if the manufacturer recommends that you use a cartridge only 10 times and you set `max_usage` to 20, reset `max_usage` to 10 if the cartridge is unused. If the cartridge was used five times, reset `max_usage` to 5.
- Set a cartridge's cleaning cartridge attribute off. For example, if you incorrectly defined a data cartridge as a cleaning cartridge, set the

cartridge's cleaning cartridge attribute off to redefine the cartridge as a data cartridge.

To change a cleaning cartridge's maximum use count:

```
set clean max_usage vol_id | volrange
```

Where:

*max\_usage* is the new maximum use.

*vol\_id | volrange* specifies the cleaning cartridge or range of cartridges.

To turn off a cartridge's cleaning cartridge attribute:

```
set clean off vol_id | volrange
```

Where *vol\_id | volrange* specifies the cartridge or range of cartridges.

## ■ Manually Loading Cartridges Into Drives In a Disabled LSM

If your LSM fails and you take it off line, you can still manually load cartridges into the library drives if the data path is still operational.

To manually load cartridges into drives in a disabled LSM:

1. Open the LSM door.
2. Note the cartridge labels of any cartridges that are already in drives and remove these cartridges.

You will need to replace these cartridges at the end of this procedure.

3. Load the drives with the cartridges you want to read from or write to.

Repeat this step as often as necessary until the LSM is repaired, then continue with Step 4.

**CAUTION:** In this step, you can remove cartridges from library cells and load these cartridges in the drives. Note the cell locations of these cartridges and ensure that you return the cartridges to these locations in Step 4.

4. After the LSM is repaired, remove all cartridges from the drives and replace them with the original cartridges you noted in Step 2.
5. Close the LSM door, vary the LSM back online, and resume normal operations.



The database contains all information about the library configuration and the location of all library cartridges. Miscellaneous files that are backed up and are recovered include the customer-configurable files located in `$ACS_Home` under `data/external` and any files associated with `data/internal/client_services`.

This chapter discusses: importing and exporting the database; verifying the imported database and library configuration; backing up the database; and restoring and recovering the database.

- Exporting and Importing the database includes:
  - Exporting the database to a disk file or local tape device
  - Importing the database from a disk file or local tape device
  - Importing miscellaneous ACSLS configuration files
  - Merging any customized dynamic variables
  - Verifying the imported database and library configuration
- Backing up the database and miscellaneous files includes:
  - Automatic database backup
  - Performing manual backups to a local tape device or to disk
  - Backing up to a specified tape device attached to the ACSLS server
  - Backing up to a UNIX File
  - Creating a backup that can be restored to a different server
- Recovering and restoring the database and miscellaneous files includes:
  - Restoring the database to the most recent backup
  - Restoring the database to a specified date and time
  - Recovering from a second disk failure
  - Recovering from a server hardware failure
  - Recovering from a specific archived file
  - Restoring miscellaneous ACSLS files
  - Restoring a backup created on a different server

- Restarting the database

## ■ Utilities Used

You will use the following utilities:

- the `bdb.access` utility for your daily backups
- the `rdb.acsss` utility for
  - recovering the database from corruption
  - from changes that produce unintended results
  - from server failure
- the `db_export.sh` and `db_import.sh` utilities for migrating between versions of ACSLS. This includes going to a later release or going to a previous release.

When you install ACSLS, you also automatically install the database management software. The ACSLS database is initialized after ACSLS is installed when you:

- configure the library hardware using `acsss_config`
- import a previous exported database using `db_import.sh`
- recover a database backup created on a different server using `rdb.acsss.sh`

## ■ Exporting the Database

This section describes how to migrate the ACSLS database and its associated miscellaneous files from either a previous version of ACSLS, the same release level of ACSLS, or return to a prior release.

The `db_export.sh` utility creates an ASCII representation of the database on tape or a specified file to disk. It is also responsible for gathering miscellaneous ACSLS configuration files. This utility can be used in two different ways.

- If it is executed without any options, the exported files are copied to the local tape drive.
- If it is executed with a “-f” option and preceded by a path and file name, the exported files reside in the name file, and its associated miscellaneous file, designated by the `.misc` extension.

The files generated by `db_export.sh` are then used as input to the `db_import.sh` utility at the time of an upgrade or recovery.

**Note:** This is the preferred method to use to migrate from all previous versions of ACSLS to the most current version.

## Exporting to a Disk File

You can export the ACSLS database and miscellaneous files to a disk file, as shown in the following procedure.

### To export the ACSLS database and miscellaneous files to a disk file:

1. Log in as `acsss`.
2. Shut down ACSLS:
  - `idle` (from `cmd_proc`)
  - `kill.acsss` (from a UNIX command prompt)

3. Shut down the database:

For ACSLS version 5.3 or 5.3.2:

```
db_command stop
```

For ACSLS version 5.4:

```
db_command.sh stop
```

For ACSLS version 6.x.x and above:

```
db_command stop
```

4. Start the `db_export.sh` utility.

```
db_export.sh -f db_file
```

You are prompted to choose the version of ACSLS that you are exporting to. Valid choices are:

- 1: 5.3.2 or 5.4
- 2: 6.0 or 6.0.1
- 3: 6.0.1 with L700e
- 4: 6.0.1 with PUT0201
- 5: 6.1, 7.0, or 7.1/7.1.1 before PUT0701
- 6: 7.1/7.1.1 with PUT0701

5. Select the desired option for the version to which you are migrating.
  - As it executes, the utility displays output indicating successful table data being exported.
  - When the export is complete, a message is displayed indicating that the export has been successful.
  - The `db_export.sh` utility creates two files: `db_file` and `db_file.misc` in the location specified with the `-f` option.

6. Ensure that these files are placed in or moved to a secure location where they won't be removed.

Do **not** put these files in or under the following directories, because these directories may be removed or deleted when ACSLS maintenance is installed:

- `$ACS_HOME`  
(the ACSSS home directory)
- `$INFORMIX_BACKUP_DIRECTORY`  
(directory where ACSLS backups are stored)
- `/tmp`

**Note:** If you plan to install a new release of the operating system, do **not** save the exported files on the ACSLS server.

## Exporting to Tape

You can export the ACSLS database and miscellaneous files to tape, as shown in the following procedure.

### To export the database and miscellaneous files to tape:

1. Log in as `acsss`.
2. Shut down ACSLS:  
`idle` (from `cmd_proc`)  
`kill.acsss` (from a UNIX command prompt)
3. Shut down the database:  
For ACSLS version 5.3 or 5.3.2:  
`db_command stop`  
For ACSLS version 5.4:  
`db_command.sh stop`  
For ACSLS version 6.x.x: and above:  
`db_command stop`
4. Insert a blank tape into the default tape device.
5. Start the `db_export.sh` utility;  
`db_export.sh`

You are prompted to choose the version of ACSLS that you are exporting to. Valid choices are:

- 1: 5.3.2 or 5.4
- 2: 6.0 or 6.0.1
- 3: 6.0.1 with L700e
- 4: 6.0.1 with PUT0201
- 5: 6.1, 7.0, or 7.1/7.1.1 before PUT0701
- 6: 7.1/7.1.1 with PUT0701

6. Select the desired option from which you are exporting.

As it executes, the utility displays output indicating successful table data being exported and successful ACSLS files being backed up. A message displays when the export is completed.

7. Remove the cartridge from the drive only when the program completes and the prompt re-appears.

**CAUTION:** You will lose files if you remove the cartridge before the program completes the export. Write protect the cartridge and clearly mark it to identify the contents as the exported database.

Do not leave the cartridge in the library.

## ■ Importing the Database

The following attributes are imported into the new database when you use the `db_import.sh` utility.

- **Volumes:** These database tables include all of the information associated with each volume in the library, such as:
  - where the volume resides
  - type of cartridge (data, scratch, cleaning, etc.)
  - last associated scratch pool
  - current status of the cartridge (home, mounted, etc.)
  - entry date and last accessed date
  - number of mounts since the entry date
  - maximum use (for cleaning cartridges)
  - associated lock ID and user ID (if the cartridge is locked)
- **ACS and Library:** database tables include the ACSs and library components, such as: LSMs, drives, panels, and cells
- **Miscellaneous:** files include all configuration updates since the initial installation, including:
  - access control information

- fixed volume preferences
- scratch media preferences
- custom volrpt templates
- Dynamic and static variables: dynamic variables that have been customized in a previous release can be imported.

This section describes how to use the `db_import.sh` utility to:

- recreate the ACSLS database
- recover important ACSLS miscellaneous files
- recover customized dynamic variables from data exported using the `db_export.sh` utility.

## Importing From a Disk File

You can import the ACSLS database and miscellaneous files from a disk file, as shown in the following procedure.

### To import the ACSLS database, miscellaneous files, or customized dynamic variables from a disk file:

1. Log in as `acsss`.
2. Shut down ACSLS:  
`idle` (from `cmd_proc`)  
`kill.acsss` (from a UNIX command prompt)
3. Shut down the database:  
`db_command stop`
4. Start the `db_import.sh` utility.  
`db_import.sh -f db_file`

The following menu displays:

```
ACSLS Import Utility
```

```
Would you like to do:
```

- 1) Import database only from either a different platform or from an earlier release
- 2) Import customizable configuration files (`$ACS_HOME/data/external` only) exported from a different platform or from an earlier release
- 3) Recover a previous environment on the same platform and version (Disaster Recovery) - database , `$ACS_HOME/data/external` and internal
- 4) Merge customized dynamic variables from a previous installation
- E) Exit

Please select one of the above:

- **Option 1 - importing only database files**

Use this option to import the database files only. This option destroys the existing database tables, rebuilds them, and then populates them with the data provided from the exported database. Output from selecting option one is similar to the following:

WARNING:

This script will destroy all tables from the ACSLS database. The results are final and there is no recovery without rebuilding the database. If you wish to preserve information in existing tables, you should not continue unless you have exported the table data using db\_export.sh.

You will need to rebuild the database tables using acsss\_config if you are not using db\_import.sh utility.

Do you wish to continue? (y or n): y

... removing ACSLS database tables.

```

Creating acstable
Creating porttable
Creating lemtable
Creating captable
Creating lockidtable
Creating drivetable
Creating volumetable
Creating celltable
Creating pooltable
Creating audittable
Creating csitable
Creating paneltable
Creating vactable
Creating scr_distr_table
Creating displaycommand table
Creating displayfields table
Creating displayoptions table
Creating displayoptval table
Creating displaysubfields table
Creating ptptable
Creating clienttable
Creating handtable
Creating imutable
db_import.sh 1657: Beginning database import phase.
db_import.sh 1658: Start copy of data into empty tables.

```

Loading Display database reference tables.

db\_import.sh 1660: Database Import Phase Complete.

ACSLs mandates backup of database in this scenario. Would you like to make an additional backup on tape? (y or n):

Selecting Y initiates an extra tape backup of the database. A backup to disk is created regardless of how you answer.

- **Option 2 - recovering ACSLS miscellaneous files**

Use this option to recover ACSLS miscellaneous files either from a previous version or the same version. This recovers all files in the directory `acs.home` under `data/external` including access control files. If access control is configured, it will also recover `data/internal/client_config`. Output is similar to the following:

```
extract_misc.sh 1663: Extracting miscellaneous ACSLS data files.  
db_import.sh 1575: Restoring miscellaneous ACSLS files...  
db_import.sh 1577: ACSLS miscellaneous files  
(/export/home/ACSSS/data/external) have been restored
```

- **Option 3 - recreating an ACSLS environment**

Use this option to recreate an ACSLS environment. This would be used:

- recovering from a hardware failure or during a hardware upgrade
- when you need to rebuild the ACSLS server to be identical to the ACSLS server from which the data was exported

This option is a combination of options 1 and 2. When finished with this option, you have the option of backing the database up to tape.

Selecting “N” at this option, performs a local disk backup.

- **Option 4 - recovering customized dynamic variables**

Use this option to recover customized dynamic variables from previous environments. This is a very useful option for upgrading versions of ACSLS without having to record previous customized dynamic variables.

Selecting this option gathers the settings from the database export, and then re-configures shared memory with the new variable settings.

**WARNING:** If you start ACSLS prior to executing this option, certain data could be lost. If you are upgrading ACSLS from a previous version and had customized dynamic variables, these changes must be made to your new environment PRIOR to starting ACSLS.

Output for this option, depending on your settings, is similar to the following:

```
extract_misc.sh 1663: Extracting miscellaneous ACSLS data files.
File
(/export/home/ACSSS/tmp/backup/data/internal/dynamic_variables/
dv_extract) is readable
Saved current configuration into file saved_cfg_20040915_1403 in
/var/tmp/saved_cfgs/
Changing the current configuration to the previously saved
configuration. Setting (TIME_FORMAT) to %Y-%m-%d)
Change complete

Configuration change complete, no errors detected.
Please review /tmp/reset_cft.tmp7.
ACSLS dynamic variables have been successfully merged.
```

5. Verify the install as described under [“Verifying the Imported Database and Library Configuration”](#) on page 214.

## Importing from Tape

Use the following procedure to import the ACSLS database, recover miscellaneous ACSLS files, and rebuild customized dynamic variables from tape.

1. Log in as `acsss`.
2. Shut down ACSLS:
 

```
idle (from cmd_proc)
kill.acsss (from a UNIX command prompt)
```
3. Shut down the database:
 

```
db_command stop
```
4. Insert the exported database tape that you exported with the `db_export.sh` command into the tape drive.
5. Run the database import utility by entering the following at a UNIX command prompt.
 

```
db_import.sh
```

The `db_import.sh` utility displays its main menu.
6. Refer to the step [“Start the db\\_import.sh utility.”](#) on page 210 for menu options.

7. Verify the install as described in [“Verifying the Imported Database and Library Configuration.”](#)

## ■ Verifying the Imported Database and Library Configuration

Use the following procedure to mount or dismount a cartridge to verify ACSLS.

### Mount/dismount a cartridge to verify ACSLS:

1. Verify that you are logged in as `acsss`.
2. If ACSLS is not running, start it by entering
3. Query the server from the `cmd_proc` by entering

```
rc.acsss
```

```
query server
```

If messages are displayed indicating that the server is in recovery mode, wait for a message indicating that the server is running.

4. Verify that the at least one of the following are online. If not, bring them online with the `vary` command.

```
query port all
```

```
query acs all
```

```
query lsm all
```

```
query drive all
```

5. Do you have at least one cartridge in an LSM?

---

YES Continue with the procedure.

---

NO Enter a cartridge into an LSM.

---

6. Mount a cartridge by entering:

```
mount vol_id drive_id
```

**Hint:** Use the `query drive` command to get the ID of an available drive and the `query volume` command to get the ID of a library cartridge. See [Chapter 14, “Command References.”](#)

7. Did you see a message indicating a successful mount?

A successful mount message is:

Mount: *vol\_id* mounted on *drive\_id*

YES	Procedure is complete.
NO	If an error message appears, run this verification procedure again, ensuring that you specified a valid, available drive and a library cartridge. If the mount/dismount still fails, call StorageTek for assistance.

8. Dismount the cartridge by entering:

```
dismount vol_id drive_id force
```

where *vol\_id* is the volume and *drive\_id* is the drive you specified in Step 6.

## ■ Automatic Database Backup

ACSLs automatically creates a backup file of the database to disk every 24 hours at midnight or the time of day and days of the week you specified in the backup options in `acsss_config`.

The automatic database backup process creates a backup of the database on the level 0 archive. Updates to the database are recorded in the logical (transaction) logs. As these logical logs fill up, they are backed up and made available for re-use.

**Table 22. Archive and Logical Logs**

Backup Type	Definition and Use
level 0 archive	<ul style="list-style-type: none"> <li>Provides a point-in-time snapshot copy of the entire database.</li> </ul>
logical logs	<ul style="list-style-type: none"> <li>Changes the database since the last backup.</li> <li>During database recovery using <code>rdb.acsss</code>, these files are applied sequentially to the point-in-time recovery to re-create the database including all transactions since the last level 0 archive.</li> </ul>

If you need to recover the database, the `rdb.acsss` utility can use the most current level 0 archive plus the logical logs to restore the database to its current state.

StorageTek supports an optional two-disk (primary and secondary) configuration for the ACSLS server. A two-disk configuration optimizes the database's ability to create backups and logical logs, and to recover from failures using these files.

In a two-disk server, the database (i.e., root database space) is mirrored. The root database space is on the primary disk, and a mirror copy is kept on the second disk. The backup is kept on the second disk in a two-disk server. For detailed procedures about database and disk recoveries, see [Appendix A, “Second Disk Support.”](#)

## ■ Performing Manual Backups to Tape

In addition to the automatic database backups that ACSLS creates, you should periodically run the `bdb.acsss` utility to manually create tape backups that can be stored offsite and used, if needed, for disaster recovery of the database. An offline backup lets you recover the database if both disks fail on a two-disk server.

**Note:** When you run the `bdb.acsss` utility, ACSLS automatically creates a backup on local disk as well. Note that running an automatic backup and `bdb.acsss` at the same time creates two local disk backups. Therefore, do not schedule automatic backups on days when you schedule `bdb.acsss`

Regular backups transferred to an offsite device can enable rapid restoration in the event of disaster to the ACSLS server.

StorageTek recommends that you use `bdb.acsss` to manually back up the database to tape after:

- Running `acsss_config`.
- Importing the database.
- An audit of the entire library.
- Any database recovery.
- Second disk installation/de-installation

## ■ Backing up to the Default Tape Device Attached to the ACSLS Server

**To back up the ACSLS database to the default tape device attached to the ACSLS server, do the following:**

Refer to [“bdb.acsss” on page 243](#).

1. Log in as `acsss`.
2. Insert a blank tape into the tape device.
3. From a terminal window, enter the following command:

```
bdb.acsss
```

4. Messages reporting the progress of the backup appear.

Wait for the following message to appear:

```
Check tape device (/dev/rmt/0) to make sure you have a  
tape in the tape drive.  
[Hit RETURN to continue or Ctrl-C to exit]
```

Press `[[Return]]`.

5. Wait for the following message:

```
ACSLS database backup successfully completed.
```

## ■ Backing up to a Specified Tape Device Attached to the ACSLS Server

To back up the ACSLS database to a specified tape device attached to the ACSLS server, do the following:

Refer to [“bdb.acsss” on page 243](#).

1. Log in as `acsss`.
2. Insert a blank tape into the tape device.
3. From a terminal window, enter the following command:

```
bdb.acsss -f tape_device
```

Where `tape_device` specifies a tape device attached to the ACSLS server.

4. Messages reporting the progress of the backup appear.

Wait for the following message to appear:

```
Check tape device (/dev/rmt/2) to make sure you have a  
tape in the tape drive.  
[Hit RETURN to continue or Ctrl-C to exit]
```

Press `[[Return]]`.

5. Wait for the following message to appear:

```
ACSLS database backup successfully completed.
```

**Example** - To back up the ACSLS database to tape device `/dev/rmt/2`, enter the following command:

```
bdb.acsss -f /dev/rmt/2
```

## ■ Backing up to a UNIX File

**Hint:** In the interest of disaster recovery, StorageTek does not recommend that you back up to a UNIX file unless the file is on a remote disk.

Refer to [“bdb.acsss” on page 243](#).

**To back up the ACSLS database to a UNIX file, do the following:**

1. Log in as `acsss`.
2. From a terminal window enter the following command:

```
bdb.acsss -f db_file
```

Where `db_file` specifies a UNIX file to contain the ACSLS database. You must have write permissions to the file.

3. Wait for the following message to appear:

```
ACSLS database backup successfully completed.
```

## ■ Backing up ACSLS and Restoring to Another Server

You can back up an ACSLS database on one server and restore it on a different but identical server. Both servers must:

- be on the same ACSLS release and maintenance level
- be on the same hardware and OS level
- have identical connections to the library hardware

**Note:** To restore a backup to another server, you must have backed up the database with the `-s` (server-to-server) option.

### Backup Process

1. Log in as user `acsss`.
2. Run `bdb.acsss -s`. The “s” option means server-to-server backup.

```
bdb.acsss -s
```

The backup process creates two files: `server_to_server.bak` and `server_to_server.bak.misc.tar.Z`. The files are found in the directory that was created during the installation process. By default, that directory is `$INFORMIX_BACKUP_DIRECTORY/server_to_server`

In most cases, the expression `$INFORMIX_BACKUP_DIRECTORY` is actually `/export/backup` or `/second_disk/backup`, depending on

whether the second disk option has been installed. If you are not sure of the actual location of this expression, type `echo $INFORMIX_BACKUP_DIRECTORY`.

3. Transfer (ftp) the newly created files, `server_to_server.bak` and `server_to_server.bak.misc.tar.Z` to a location external to the server being backed up.

## Restore the Backup to a Different Server

1. Log in as `acsss` to the machine where you will do the restore.
2. Transfer (ftp) the files from the archive location to the directory where you are restoring the files.

You need both `server_to_server.bak` and `server_to_server.bak.misc.tar.Z`. They need to be placed in the `server_to_server` directory (`$INFORMIX_BACKUP_DIRECTORY/server_to_server`).

The files need to be owned by the user *informix* and have group ownerships of *informix*. If they are not, you need to:

- login as `root`
- `+ chown informix /export/backup/server_backup_file`
- `+ chgrp informix /export/backup/server_backup_file`

3. Bring down ACSLS.  
`kill.acsss`
4. Bring down the Informix Database Server.  
`db_command stop`
5. Initiate a database restore  
`rdb.acsss`
6. Select option number 4 “Restore a backup created on a different server”
7. A prompt similar to the following is displayed:  

```
Please mount tape 1 on /export/backup/  
server_backup_file and press Enter to continue ...
```

  
This is the file transferred from the server on which the back up was created.
8. Press <Enter>  
  
Since you are backing up from a file on disk, you can disregard the suggestion to “mount a tape”.
9. An Informix prompt similar to the following is displayed:

```
Continue restore? (y/n)
```

10. Type y for yes.

11. A prompt similar to the following is displayed:

```
Do you want to back up the logs? (y/n)
```

You do not need to backup the logical logs.

12. Type n for no.

13. A prompt similar to the following is displayed:

```
Restore a level 1 archive (y/n)
```

Since you are restoring a level 0 backup, there is no need to restore a level 1 backup.

14. Type n for no.

15. A prompt similar to the following is displayed:

```
Do you want to restore log tapes? (y/n)
```

We are not restoring the logical logs.

16. Type n for no.

The restore is complete.

17. Select option 6 to exit.

A prompt similar to the following is displayed:

```
After performing a successful Informix database recovery you should do a disaster recovery backup. This will ensure your ability to recover again should you have another database failure. Do you want to do this database backup now? (y or n):
```

- If you respond y, an automatic backup is made first to local disk. You see several messages indicating the backup has started, followed by messages showing the backup successfully completed. The backup is made to tape. If you do not have a tape in the tape drive, you are prompted to put one in.
- If you respond n, an automatic backup is made to local disk. You see a series of messages indicating the backup has started, followed by messages showing the backup completed successfully.

## ■ Recovering and Restoring

This section describes the following restoration/recovery procedures:

- Restoring a corrupted or lost database to the most recent backup
- Restoring a corrupted or lost database to a specified date and time
- Recovering from a primary disk failure
- Recovering from a secondary disk failure
- Disaster recovery for a failed server
- Recovering from a specific archived file
- Restoring non-database, miscellaneous files

Most of these procedures use the `rdb.acsss` utility, which provides options for restoring a database from the most recent backup or from a specified date and time; disaster recovery using a backup created by `bdb.acsss`; and restoring miscellaneous files created by `bdb.acsss`. For more information about these options, see [“rdb.acsss” on page 275](#).

The `rdb.acsss` utility loads the database backup, then sequentially applies any available transaction log files to restore the database. If the transaction log files are available, the database can be restored to its state just before the failure, with essentially no loss of data.

**Note:** If the home cell of a cartridge changes from its last location after a backup, then the restored database will not be up-to-date. To avoid cartridge movement on dismounts:

- each LSM must be the only LSM in its ACS (true in most SCSI libraries), or
- the Extended Store Feature must be enabled for all LSMs that are connected to other LSMs via a pass-thru-port.

(For more information, see [“Using the Extended Store Feature” on page 167](#)). If the Extended Store Feature is not enabled for all connected LSMs or cartridges have been entered or ejected, you must audit the library after the restoration to make the database current.

**Note:** Do not specify the `-f` option as a general option for the `rdb.acsss` utility. If you backed up your database to an external network file or to an alternate tape device, you use the `-f` option only after entering `rdb.acsss` and then choosing the third recovery option. When prompted, enter `-f` and the path name to your external network file or alternate tape device. See [“Select the third option:” on page 228](#) for more information.

## ■ Restoring the Database to the Most Recent Backup

In this procedure you restore the database to the most recent backup created on the local disk by automatic backups. Transaction log files are applied to make the database as current as possible. Miscellaneous ACSLS files from the backup are restored.

### To restore a corrupted or lost database to the most recent backup:

1. Log in as `acsss`.
2. From the `cmd_proc` window, idle ACSLS:

```
idle
```

3. Shut down ACSLS:

```
kill.acsss
```

4. Shut down the database:

For ACSLS version 6.x.x:

```
db_command stop
```

5. Enter the following command:

```
rdb.acsss
```

The Database Recovery menu displays six options. For more information about these options, see [“rdb.acsss” on page 275](#).

6. Select the first option:

```
1. Restore from current local disk backup
```

7. Respond `n` to the following prompt:

```
Are you restoring the primary disk from the second
disk? (This will overlay the current version of the
ACSLs and Informix configuration files.) (y or n):
```

You should see messages indicating the recovery is in progress and the recovery completed successfully.

The Database Recovery menu displays.

8. Select option 6 to exit the recovery utility.

You see the following prompt:

```
After performing a successful Informix database
recovery you should do a disaster recovery backup. This
will ensure your ability to recover again should you
```

have another database failure. Do you want to do this database backup now? (y or n):

9. Respond y or n.

If you respond y, an automatic backup is made first to local disk. You see several messages indicating the backup has started, followed by messages showing the backup successfully completed. Then the backup is made to tape. If you do not have a tape in the tape drive, you are prompted to put one in.

If you respond n, an automatic backup is made to local disk. You see a series of messages indicating the backup has started, followed by messages showing the backup successfully completed.

10. To start ACSLS, enter the following command:

```
rc.acsss
```

## ■ Restoring the Database to a Specified Date and Time

In this procedure you restore the database to a backup prior to a specified date and time within the backup retention period. Archived transaction log files are applied to the backup up to the specified time. Miscellaneous ACSLS files are restored only from the backup.

### To restore a database to a specified date and time:

1. Log in as `acsss`.
2. From the `cmd_proc` window, idle ACSLS:

```
idle
```

3. Shut down ACSLS:

```
kill.acsss
```

4. Shut down the database:

For ACSLS version 6.x.x:

```
db_command stop
```

5. Enter the following command:

```
rdb.acsss
```

The Database Recovery menu displays five restoration options. For more information about these options, see [“rdb.acsss” on page 275](#).

6. Select the second option:

2. Restore from a previous local disk backup (to a point in time)

7. The following prompt displays:

```
Informix database recovery started.  
You have taken backups on the following days. Please  
enter any date and time after the earliest backup  
displayed. The database will be recovered to the date  
and time you specified.  
YYYY-MM-DDHH:MM:SS  
YYYY-MM-DDHH:MM:SS  
YYYY-MM-DDHH:MM:SS
```

```
Please enter the recovery date and time (YYYY-MM-DD  
HH:MM:SS):
```

8. Enter the date and time to which you want to recover the database.

You should see messages indicating the recovery is in progress and the recovery completed successfully.



**CAUTION:** If you previously did a point-in-time restoration, you can do a later point-in-time restoration *only* when the time selected is **before** the time specified for the earlier restoration.

9. Respond y to the following prompt:

```
Do you want to restore the ACSLS and Informix  
configuration files to their previous state? (y or n):
```

You should see messages indicating the recovery of miscellaneous and configuration files.

10. When the Database Recovery menu appears, select option 6 to exit the recovery utility.

You see the following prompt:

```
After performing a successful Informix database  
recovery you should do a disaster recovery backup. This  
will ensure your ability to recover again should you  
have another database failure. Do you want to do this  
database backup now? (y or n):
```

11. Respond y or n .

If you respond y, an automatic backup is made first to local disk. You see several messages indicating the backup has started, followed by messages showing the backup successfully completed. Then the backup is made to tape. If you do not have a tape in the tape drive, you are prompted to put one in.

If you respond `n`, an automatic backup is made to local disk. You see a series of messages indicating the backup has started, followed by messages showing the backup successfully completed.

- To start ACSLS, enter the following command:

```
rc.acsss
```

## ■ Recovering from a Primary Disk Failure

In this procedure, you recover a corrupted or lost primary disk from the secondary disk. To use this procedure, the secondary disk must be fully operational.

### To recover from a primary disk failure:

- If necessary, install the new primary disk. Follow the manufacturer's instructions.
- Install the operating system.
- Install ACSLS.



**CAUTION:** You *must* install ACSLS in the same directory you used before the disk failure.

- Log in as `acsss`.
- From the `cmd_proc` window idle ACSLS:

```
idle
```

- Shut down ACSLS:

```
kill.acsss
```

- Shut down the database:

For ACSLS version 6.x.x:

```
db_command stop
```

- Enter the following command:

```
rdb.acsss
```

The Database Recovery menu displays 6 restoration options. For more information about these options, see [“rdb.acsss” on page 275](#).

- Select the first option:

```
1. Restore from current local disk backup
```

- Respond `y` to the following prompt:

Are you restoring the primary disk from the second disk? (This will overlay the current version of the ACSLS and Informix configuration files.) (y or n):

11. Press <Enter> at the following prompt if you were using /second\_disk directory for second disk support:

What directory were you using for second disk support [/second\_disk]?

12. Press <Enter> at the following prompt if you were using /second\_disk/backup directory for second disk backups:

What second disk directory were you using for Informix backups [/second\_disk/backup]?

13. You see several messages indicating the recovery is in progress and the recovery completed successfully. When the Database Recovery menu appears, select option 6 to exit the recovery utility.

You see the following prompt:

After performing a successful Informix database recovery you should do a disaster recovery backup. This will ensure your ability to recover again should you have another database failure. Do you want to do this database backup now? (y or n):

14. Respond y or n.

If you respond y, an automatic backup is made first to local disk. You see several messages indicating the backup has started, followed by messages showing the backup successfully completed. Then the backup is made to tape. If you do not have a tape in the tape drive, you are prompted to put one in.

If you respond n, an automatic backup is made to local disk. You see a series of messages indicating the backup has started, followed by messages showing the backup successfully completed.

15. To start ACSLS, enter the following command:

```
rc.acsss
```

16. You must run acsss\_config to re-specify automated backup date and time and retention periods (Selection 5) *unless* you want to accept the default settings.

## ■ Recovering from a Second Disk Failure

Use this procedure if you have lost or corrupted the second disk.

**To recover from a second disk failure:**

1. De-install second disk support.

Follow the procedures in [Appendix A, “Second Disk Support.”](#)

You can now run on just the primary disk.

**Hint:** While you are using just the primary disk, you should take frequent backups to tape or to external disk using the `bdb.acsss` utility. See [“bdb.acsss” on page 243.](#)

2. If necessary, remove the second disk and re-install a new second disk following the manufacturer's instructions.
3. Install second disk support.

## ■ Recovering from a Failed Server

Use this procedure for a disaster recovery when you have lost or corrupted both primary and secondary disks.

### To recover from a failed server:

1. If necessary, de-install both the primary and secondary disks and re-install new primary and secondary disks, following the manufacturer's instructions.
2. Install the operating system.
3. Install ACSLS.



**CAUTION:** You *must* install ACSLS in the same directory you used before the disk failure.

4. Log in as `acsss`.
5. From the `cmd_proc` window idle ACSLS:

```
idle
```

6. Shut down ACSLS:

```
kill.acsss
```

7. Shut down the database:

For ACSLS version 6.x.x:

```
db_command stop
```

8. Enter the following command:

```
rdb.acsss
```

The Database Recovery menu displays 6 restoration options. For more information about these options, see [“rdb.acsss” on page 275.](#)

9. Select the third option:

3. Restore from a previous tape or network file backup

This option restores the database from a backup created by `bdb.acsss`.

The following prompt displays:

To recover the ACSLS environment either:

- Mount an ACSLS backup tape in the default tape device and press Enter,
- Mount an ACSLS backup tape in another tape device and specify this tape device with '-f tape\_device', or
- Specify an external (network) file name containing an ACSLS backup with '-f backup\_file'.

Please mount tape (if used) and enter backup source:

10. Accept the default tape device by pressing <Enter>, or enter an external network filename or an alternate tape device by entering:

`-f pathname`

After you respond to this prompt, you see several messages indicating the recovery is in progress and the recovery completed successfully.

11. When the Database Recovery menu appears, select option 6 to exit the recovery utility.

You see the following prompt:

After performing a successful Informix database recovery you should do a disaster recovery backup. This will ensure your ability to recover again should you have another database failure. Do you want to do this database backup now? (y or n):

12. Respond y or n.

If you respond y, an automatic backup is made first to local disk. You see several messages indicating the backup has started, followed by messages showing the backup successfully completed. Then the backup is made to tape. If you do not have a tape in the tape drive, you are prompted to put one in.

If you respond n, an automatic backup is made to local disk. You see a series of messages indicating the backup has started, followed by messages showing the backup successfully completed.

13. To start ACSLS, enter the following command:

`rc.acsss`

14. You must run `acsss_config` to re-specify automated backup date and time and retention periods (Selection 5) *unless* you want to accept the default settings.

## ■ Recovering from a Specific Archived File

There may be contexts in which you would like to recover from a specific tape or disk file archive that you have previously taken using `bdb.acsss -f`. You can use a specific backup file for recovery under two conditions:

- The archived backup that is used in recovery is the most *recent* backup taken on the system. In this case, do the following:
  1. Run the `bdb.acsss` utility.
  2. Select Option 3, using `-f` to specify the backup file or tape device.
- The archived backup is used immediately after re-installing the Informix `/export/home` and `/export/backup` directories. In this case, do the following:

1. As root user, remove the following directories:

```
/export/home/informix
```

```
/export/backup/informix
```

or

```
/second_disk/backup informix
```

```
/export/backup/misc
```

or

```
/second_disk/backup/misc
```

2. Reinstall Informix from the ACSLS CD-ROM.

- a. Insert the CD-ROM into the CD-ROM drive.
- b. Execute the `install.sh` script.

The installation script will replace only the directories that were removed in Step 1.

3. Restore the database

- a. Run the `bdb.acsss` utility.
- b. Select Option 3, using `-f` to specify the backup file or tape device.

## ■ Restoring Miscellaneous ACSLS Files

In this procedure you restore miscellaneous ACSLS files. These are non-database files that include all files in the `data/external` directory such as

access control files, the fixed volume file, the scratch preferences file, and custom volrpt files. These files are restored from a bdb.acsss backup to tape or an external network file.

**To restore miscellaneous ACSLS files:**

1. Log in as `acsss`.
2. From the `cmd_proc` window, idle ACSLS:

```
idle
```

3. Shut down ACSLS:

```
kill.acsss
```

4. Shut down the database:

For ACSLS version 6.x.x:

```
db_command stop
```

5. Enter the following command:

```
rdb.acsss
```

The Database Recovery menu displays 6 restoration options. For more information about these options, see [“rdb.acsss” on page 275](#).

6. Select option 5:

```
4. Restore ACSLS non-database files
```

7. When the following prompt appears, accept the default tape device or enter an external network filename or an alternate tape device:

To recover the ACSLS environment either:

- Mount an ACSLS backup tape in the default tape device and press Enter,
- Mount an ACSLS backup tape in another tape device and specify this tape device with `'-f tape_device'`, or
- Specify an external (network) file name containing an ACSLS backup with `'-f backup_file'`.

Please mount tape (if used) and enter backup source:

After you respond to the prompt, you see several messages indicating the recovery is in progress and the recovery completed successfully.

8. When the Database Recovery menu appears, select option 6 to exit the recovery utility.

You see the following prompt:

After performing a successful Informix database recovery you should do a disaster recovery backup. This

will ensure your ability to recover again should you have another database failure. Do you want to do this database backup now? (y or n):

9. Respond y or n.

If you respond y, an automatic backup is made first to local disk. You see several messages indicating the backup has started, followed by messages showing the backup successfully completed. Then the backup is made to tape. If you do not have a tape in the tape drive, you are prompted to put one in.

If you respond n, an automatic backup is made to local disk. You see a series of messages indicating the backup has started, followed by messages showing the backup successfully completed.

10. To start ACSLS, enter the following command:

```
rc.acsss
```

## ■ Restarting the Database

Use this procedure to start up the database manually. Normally, the database is started automatically when the library server is booted. Typically, you would use this procedure if the database is shut down but ACSLS is still running (for example, during a recovery).

### **To restart the database, do the following:**

1. Log in as `acsss` and open a UNIX command tool.
2. Enter the following command:

```
db_command start
```



In this chapter, you will learn how to:

- Create a custom volume report
- Create a report for volume movement statistics

## ■ Creating a Custom Volume Report

You can use the `volrpt` utility to create a volume report; for more information, see the “[volrpt](#)” on page 281. `$ACS_HOME/data/external/volrpt/owner_id.volrpt` is a sample input file that you can run or use as a template to create customized volume reports. You can also save your customized volume reports in the `$ACS_HOME/data/external/volrpt` directory. Use this procedure to create an input file to the `volrpt` utility to create a custom volume report.

**To create a custom volume report, do the following:**

1. Log in as `acsss`.
2. Open a UNIX command tool.
3. Change to the custom volume report directory:  

```
cd /home/ACSSS/data/external/volrpt
```
4. Copy the sample volume report file to a new, custom file.  

```
cp owner_id.volrpt my.volrpt
```

*my.volrpt* is the name you want to assign to the new file.
5. Using a text editor, such as `vi`, edit the *my.volrpt* file to specify the fields and formats you want to see on the custom report.
  - You can specify any of the fields listed in the sample file.
  - The format for each entry is:  
`field_name field_length delimiter_length`
  - You can make the field lengths and delimiters any size you wish. Just be sure all the fields you specify will fit on one line when the report prints.
  - Detailed editing instructions are given in the sample file.
6. When you are finished editing the file, save it.

## Custom Volume Report

You create a customized report by designating fields, field length, and delimiter lengths in an input file, such as the one shown in [Figure 38](#):

**Figure 38. Sample Volrpt Input File**

```
#####
#
# File name: owner_id.volrpt
#
# This file describes the report layout for volrpt invoked with the -f option.
#   volrpt -f <filename>
#
# The format of a line is:
#   field_name      field_length      delimiter_length
#
# The field length is the number of characters which will be printed for
# the field. The delimiter length is the number of spaces that will be
# printed after the field. If you leave out the lengths, or specify a
# value of -1, the default values will be used. Default delimiters are
# always 2. Here are the fields and their default lengths.
#
# ACCESS_COUNT      5  2
# ACCESS_DATE       15 2
# CELL_ID           13 2
# DRIVE_ID          10 2
# ENTRY_DATE        15 2
# LABEL_ATTR        5  2
# LOCK_ID           5  2
# LOCK_TIME         15 2
# MAX_USE           5  2
# MEDIA_TYPE        7  2
# OWNER_ID          20 2
# POOL_ID           5  2
# VOLUME_ID         6  2
# VOL_STATUS        17 2
# VOLUME_TYPE       4  2
#
# Revision History:
# xx/xx/xx  Name      Changes
#
#####
VOLUME_ID          6  2
MEDIA_TYPE         7  2
DRIVE_ID           12 2
CELL_ID            15 2
OWNER_ID           -1  0
```

From this input file you generate the customized report shown. Among other uses, a customized report ([Figure 39](#)) allows you to report the owners of volumes.

**Figure 39. Customized Volume Report Using Input File**

```
1998-06-30 13:22:07

TOTAL VOLUMES: 2      SEQUENCE:      sort by volume identifier

Volume Media          Drive ID      Home          Owner
Label  Type                               Location     ID

RB1400 3480           Not-in-drv   0, 1, 1, 0, 0  cray
RB1401 DD3A           0, 0, 1, 0   0, 1, 2, 0, 0  cray
```

## Custom Volume Report Example 2

The following volrpt includes ACSs 0 and 1, absent and ejected volumes, and shows the status of volumes ([Figure 40](#) and [Figure 41 on page 236](#)).

The input volrpt options are:

```
volrpt -f my.volrpt -a 0 1 -i
```

These control statements are used to select and format the output:

**Figure 40. Sample volrpt Control Statements**

CELL_ID	13	2
VOLUME_ID	6	2
VOL_STATUS	17	0
POOL_ID	5	2
ACCESS_COUNT	5	1
LOCK_ID	5	1
OWNER_ID	20	0

**Figure 41. Customized volrpt Showing Absent and Ejected Volumes**

```

VOLUME REPORT UTILITY
2002-06-03 15:27:48

TOTAL VOLUMES: 61      SEQUENCE: sort by volume identifier

```

Home Location---	Volume Label	Volume Status	Pool ID	Times Mount	Lock ID	Owner ID-----
1, 0, 0, 0, 0	ABC001	VOLUME_HOME	0	2	0	presc
0,-1, 0, 0, 0	ABC002	VOLUME_ABSENT	0	0	0	
0,-1, 0, 0, 0	ABC003	VOLUME_ABSENT	0	0	0	
1, 3, 0, 0, 3	ABC004	VOLUME_MISSING	0	0	0	
1, 3, 0, 0, 4	ABC005	VOLUME_MISSING	4	0	28001	tom
1, 3, 0, 0, 5	ABC006	VOLUME_MISSING	0	0	0	
0,-1, 0, 0, 0	ABC007	VOLUME_ABSENT	0	0	0	
1, 0, 0, 0, 7	ABC008	VOLUME_HOME	0	0	0	
0,-1, 0, 0, 0	ABC009	VOLUME_ABSENT	0	0	0	
0,-1, 0, 0, 0	ABC010	VOLUME_ABSENT	0	0	0	presc
1, 0, 0, 0,10	ABC011	VOLUME_HOME	0	0	0	
1, 0, 0, 0,12	ABC012	VOLUME_HOME	0	0	2371	abc012
1, 0, 0, 0,13	ABC013	VOLUME_HOME	0	0	28001	
1, 0, 0, 0,14	ABC014	VOLUME_HOME	0	0	28001	
0,-1, 0, 0, 0	ABC015	VOLUME_ABSENT	1	0	29712	
0,-1, 0, 0, 0	ABC016	VOLUME_EJECTED	1	0	29712	
0,-1, 0, 0, 0	ABC017	VOLUME_ABSENT	1	0	29712	
0,-1, 0, 0, 0	ABC018	VOLUME_ABSENT	1	0	29712	
1, 0, 0, 0,19	ABC019	VOLUME_HOME	1	0	0	
1, 0, 0, 0,20	ABC020	VOLUME_HOME	1	0	0	
0,-1, 0, 0, 0	ABC021	VOLUME_ABSENT	0	0	0	
0,-1, 0, 0, 0	ABC022	VOLUME_ABSENT	4	0	0	

## ■ Creating a Logging Volume Statistics Report

You can use the volume statistics log file (`acsss_stats.log`) to log volume movement statistics (Figure 42 on page 237). These statistics consist of entries for each time ACSLS detects that a volume's location changed. ACSLS logs entries for enters, ejects, mounts, dismounts, and for each time an audit detects that a volume's location has changed (typically, by being manually moved).

You use the `acsss_config` configuration program to do the following:

- Enable or disable volume statistics logging via the `LIB_VOL_STATS` variable
- Specify the maximum size of the volume statistics log file
- Specify the number of rollover files for the volume statistics log file

The `stats_report` utility uses the `acsss_stats.log` to report all mounts and tape drive usage

The volume statistics log file contains collection mode entries that tell if volume statistics logging is enabled or disabled and *volume statistics entries*.

**Figure 42. Example of Volume Statistics Log File Entries**

```

1998-06-30 08:53:00 CONFIG
Library volume statistics on.

1998-06-30 09:23:08 EJECT
U01120 Home 0,0,1,3,5 Cap 1,0,0 Client Host Id 129.81.15.25

1998-06-30 10:36:05 ENTER
PB0444 Home 0,0,4,3,5 Cap 0,0,0 Client Host Id 129.81.15.25

1998-06-30 10:42:48 MOUNT
PB0478 Home 0,0,1,35,1 Drive 0,0,1,0 Client Host Id Local

1998-06-30 10:43:19 DISMOUNT
PB0478 Home 0,0,1,35,1 Drive 0,0,1,0 Client Host Id Local

1998-06-30 10:43:19 AUDIT
RB0478 0,0,1,35,1 STATUS_VOLUME_NOT_FOUND Client Host Id JBHUTTO

1998-06-30 10:43:19 AUDIT
PB0444 0,0,1,32,1 STATUS_VOLUME_FOUND Client Host Id JBHUTTO

1998-06-30 10:45:00 CONFIG
Library volume statistics off.

2001-01-16 09:51:07 ACSCR
0A1235 Home 0,0,5,14,14 STATUS_VOLUME_NOT_FOUND Client Host Id Local

2001-01-16 09:40:13 ACSCR
0A123A Home 0,0,5,14,15 STATUS_VOLUME_FOUND Client Host Id Local

```

In [Figure 42](#), the collection mode entries show that statistics collection started at 8:53 a. m. on May 30, 1998 and ended at 10:45 a.m. the same day. These collection start and stop times bracket the volume statistics entries for this collection period.

The format of the volume statistics entries is:

```

yyyy-mm-dd hh:mm:ss command
vol_id home_loc function_loc client_host_ID

```

Where:

*yyyy-mm-dd*

is the year, month, and day of the entry. Four-digit year formats are supported. You specify the date format via the `acsss_config` program.

*hh:mm:ss*

is the hour, minute, and second of the entry.

*command*

is the ACSLS command or client request that either moved the volume or (for audits) detected that the volume was moved.

MOUNT  
mount request.

DISMOUNT  
dismount request.

ENTER  
manual or automatic mode enter request.

EJECT  
eject request.

AUDIT  
audit request.

ACSCR  
Volume Recovery activity. This activity is generated automatically by ACSLS processing.

*vol\_id*

is the volume IDs

*home\_loc*

is the volume's home (storage cell) location.

*function\_loc*

is the volume's location for the requests that used the volume as follows:

mount or dismount requests  
location is a transport ID.

enter or eject requests  
location is a CAP ID.

audit requests

specifies that an audit detected one of the following errors:

STATUS\_VOLUME\_FOUND

The audit found a volume in a location that did not match the location specified in the database.

STATUS\_VOLUME\_NOT\_FOUND

The audit did not find a volume in the location specified in the database.

Volume Recovery activity

Location may be a cell ID or a transport ID, indicating that Volume Recovery detected one of the following situations:

STATUS\_VOLUME\_FOUND

A volume was found which was not recorded in the database, and is being added.

STATUS\_VOLUME\_NOT\_FOUND

A volume in the database was not found in any recorded location, and is being deleted.

*client\_host\_ID*

is one of the following;

- For client application requests, the host IP address.
- For `cmd_proc` commands, if the environment variable `LIBVOLSTATS_CMD_PROC_ID` is set (ASCII characters only) in the environment of the shell that started the `cmd_proc`, the entry is the first 12 characters of the value of the environment variable.
- For `cmd_proc` commands, if the environment variable `LIBVOLSTATS_CMD_PROC_ID` is not set or contains non-ASCII characters, the entry is `Local`.



---

In this chapter, you will learn how to use the following ACSLS utilities:

- “[bdb.acsss](#)” on page 243  
backs up the ACSLS database and miscellaneous library resource files.
- “[cmd\\_proc\\_shell](#)” on page 245  
provides shell access to the `cmd_proc`.
- “[Dynamic Configuration \(config\) utilities](#)” on page 245  
dynamically implements configuration changes to ACSLS libraries (and components) while ACSLS remains online and running. These configuration utilities are:
  - “[config acs](#)” on page 247  
dynamically adds an ACS or re-configures an existing ACS and its components.
  - “[config drives](#)” on page 249  
on existing drive panels, it dynamically adds drives, changes drive types, and deletes drives.
  - “[config lsm](#)” on page 250  
dynamically re-configures an existing LSM and all of its components. These components include CAPs, panels, and drives.
  - “[config ports](#)” on page 251  
dynamically re-configures the port connections to an ACS.
  - “[db\\_command](#)” on page 252  
starts or stops the database.
  - “[db\\_export.sh](#)” on page 253  
exports the ACSLS database information and miscellaneous files in preparation for an upgrade installation or reinstallation of ACSLS.
  - “[db\\_import.sh](#)” on page 254  
imports the ACSLS database information and miscellaneous files exported when you used the `db_export.sh` utility.
  - “[del\\_vol](#)” on page 254  
deletes a volume from an offline LSM.
  - “[drives\\_media.sh](#)” on page 256

displays all drive types, media types and the drive-to-media compatibilities that are supported by the current release of ACSLS.

[“ejecting.sh” on page 260](#)

determines a list of volumes that are to be ejected and creates a file containing the set of eject commands for the specified CAP.

[“find\\_tty.sh” on page 261](#)

identifies any available port that can be used by acsss.

[“free\\_cells.sh” on page 262](#)

allows you to monitor and manage the free cells in libraries controlled by ACSLS.

[“get\\_license\\_info” on page 265](#)

displays details about the installed ACSLS license. This includes the number of volumes and free cells currently supported by ACSLS.

[“greplog” on page 266](#)

filters the acsss\_event log to include or exclude messages containing specific keywords.

[“kill.acsss” on page 267](#)

terminates ACSLS.

[“moving.sh” on page 267](#)

moves multiple cartridges to one or more LSMs.

[“probeFibre.sh” on page 274](#)

displays the model number, revision level, and Target-LUN address of each device connected behind an Emulex (LP10000) or QLogic (QLA2300) fibre-channel HBA.

[“rc.acsss” on page 274](#)

starts ACSLS.

[“rdb.acsss” on page 275](#)

restores the ACSLS database and miscellaneous library resource files.

[“showDevs.sh” on page 278](#)

shows detail for every mchanger device configured on Solaris.

[“stats\\_report” on page 280](#)

gathers library volume statistical information.

[“volrpt” on page 281](#)

creates a volume report.

[“watch\\_vols” on page 286](#)

automatically assigns ownership and pool association to volumes as they are entered through the CAP.

## ■ Using the ACSLS Utilities

Follow these general guidelines for using the ACSLS utilities:

- You must log in as `acsss` to ensure the proper environment to run ACSLS utilities.

You should not `su` to `acsss`.

- StorageTek recommends that you use `bdb.acsss` to manually back up the database to tape after:
  - Configuring your library hardware.
  - Importing the database. After you upgrade to a new version of ACSLS, do not use database backups or exports created with previous versions. Make sure, however, to create a new backup as soon as you have upgraded.
  - Any database recovery.
- To ensure that you recover an accurate and consistent database, always use the most current database backup.

If a utility fails, retain all event logs. These logs will aid StorageTek in resolving any problems. For more information, see [“Logs of ACSLS Installation and Utilities” on page 508](#)

## ■ `bdb.acsss`

The `bdb.acsss` utility backs up the ACSLS database and miscellaneous library resource files.

**Note:** When you run the `bdb.acsss` utility, ACSLS automatically creates a backup on local disk as well. Note that running an automatic backup and `bdb.acsss` at the same time creates two local disk backups. Therefore, do not schedule automatic backups on days when you schedule `bdb.acsss`.

### Format

```
bdb.acsss [-s] [-f db_file | tape_device]
```

**Hint:** If you enter `bdb.acsss` with no options, the backup utility defaults to the default tape device attached and configured to the ACSLS server.

### Options

**Note:** `[-s]` and `[-f]` are mutually exclusive options.

`[-s]`

creates a server-to-server backup. Locates the backup on a file system external to the ACSLS server.

**Note:** Make sure this location has space for a large backup. This file could be as large as 200 megabytes.

-f *db\_file*

specifies a UNIX file to contain the ACSLS database backup. You must have write permissions to the file.

-f *tape\_device*

specifies any tape device attached and configured to the ACSLS server.

## Usage

Use the `bdb.acsss` utility to manually back up the ACSLS database to tape or to an external network file to create backups that can be used, if needed, for disaster recovery of the database. A tape backup, for example, lets you recover the database if both disks fail on a two-disk server. See [“Performing Manual Backups to Tape” on page 216](#) for detailed procedures for different types of backups.

How often you back up the database depends on the amount of library activity. In general, you should back up the database often for frequent changes to library contents or configuration.

StorageTek recommends that you use `bdb.acsss` to manually back up the database to tape after:

- Running `acsss_config`.
- Importing the database. After you upgrade to a new version of ACSLS, *do not* use database backups or exports created with previous versions. Make sure, however, to create a new backup as soon as you have upgraded.
- An audit of the entire library.
- Any database recovery.

**Note:** Recovery from a manual backup will succeed under either of the following two conditions:

- The archived backup that is used in recovery is the most recent backup taken on the system.
- The archived backup is used immediately after re-installing the Informix directories in `/export/home` and `/export/backup`.

See [“Recovering from a Failed Server” on page 227](#) and [“Recovering from a Specific Archived File” on page 229](#).

See also:

- [“Performing Manual Backups to Tape” on page 216](#)
- [“rdb.acsss” on page 275](#)
- [Chapter 11, “Database Backup and Restore,” backup sections and “Recovering and Restoring” on page 220](#)

## ■ cmd\_proc\_shell

This utility allows you to run `cmd_proc` from a shell in a window with a scroll bar with the command output no longer scrolling off the screen. It inherits all of the attributes of a given shell, including command history, command editing, and scrolling.

### Format

```
cmd_proc_shell <ACSL Command>
```

## ■ Dynamic Configuration (config) utilities

The dynamic configuration (`config`) utility allows you to implement configuration changes to ACSLS libraries (and components) while ACSLS remains online and running. These configuration changes are recorded in the `acsss_config.log` file.

The following dynamic configuration utilities are supported:

- `config acs`
- `config drives`
- `config lsm`
- `config ports`

Using the `config` utility provides the following benefits:

- ACSLS can continue running, allowing you to perform mount requests to unaffected library components.
- Allows you to reconfigure specified library components while all other configuration information remains unchanged. For example, when specifying:
  - An ACS, the configurations of other ACSs are not affected.
  - An LSM, the configurations of other LSMs are not affected.
  - A drive panel, the drives on a panel, mounts and dismounts to all existing drives are not affected.

## Important

- ACSLS must be running to use the dynamic `config` utility.
- You will need to enter a new licence key if you have increased the number of storage cells in your existing configuration and that increase now exceeds your current licensed capacity.

You will then need to perform an audit of the library. This will update the ACSLS database to match the actual inventory of library cartridges.

- You must use `acsss_config` to create your initial ACSLS configuration. Refer to [“ACSL Configuration” on page 115](#).
- Event notification reports all dynamic configuration changes.
- Before running dynamic configuration, ensure that all components being added or re configured are ready.
- The `acsss_config.log` file provides details regarding messages displayed.
- If you have not confirmed the configuration change, you can cancel the operation with `[[CTRL]]+C`.
- Dynamic configuration performs an automatic backup before and after the configuration change.
- After a configuration change is confirmed, it cannot be cancelled. If you want to reverse a configuration change, you should shut down ACSLS and restore the backup that was taken immediately before the configuration change was made.

You have 10 minutes to confirm a configuration change before it times out.

- You cannot remove the only (or last) ACS.

## Limitations of Dynamic Configuration

The dynamic configuration utility has two significant limitations:

- You cannot delete an ACS, or delete or change a port (connection) to a library.
- With an existing SCSI/fibre-attached library, you can **only update** drive configurations. The `config drives` utility works without limitations. The `config acs` and `config lsm` utility will only update the drives in a SCSI/fibre-attached library. You cannot update panel cell or CAP configurations in a SCSI/fibre-attached library. `config acs new` works for SCSI/fibre-attached libraries.

**Solution:**

For these configuration changes that are not supported through dynamic configuration, you need to shutdown ACSLS and use `acsss_config`.

**Things You Should Not Do**

- Do **not** use this utility to display status information for a library and its components.

Since it involves extensive I/O to the library, you need to use the `query` or `display` commands.

- Try to perform more than one configuration task at a time.

Only one dynamic configuration task can be performed at a time. This:

- Minimizes performance problems caused by the I/O between ACSLS and the library it is configuring,
- Avoids complex interactions between multiple configuration tasks.

**■ config acs**

The `config acs` utility allows you to:

- Add an ACS or re-configure an existing ACS and its components
- Configure a partitioned ACS
- Add or remove LSMs, since the `config lsm` utility only allows you to reconfigure an existing LSM.

**Format**

To add a new ACS, enter:

```
config acs new
```

To reconfigure an existing ACS, enter:

```
config acs acs_id
```

**Adding a new ACS**

To add a new ACS:

1. Enter:

```
config acs new
```

2. Enter the number of connections to the ACS followed by the device or host ID(s).

**CAUTION: Ensure that all ports are connected to the same ACS.**

The next available number in the list of existing ACSs is assigned to the new ACS. For example, if your existing ACSs are numbered 0, 1, 3, 4, the new ACS would be assigned the number 2.

3. If this is an SL8500 library, enter the partition ID for the ACS.

This must match the partition ID on the SLConsole. If this is not a partitioned library, enter **0**.

The new ACS configuration is displayed.

4. Confirm the addition of the new ACS .

After confirmation, the configuration information is displayed and the database is updated.

## Reconfiguring an existing ACS

The ACS should, if possible, be online or in diagnostic mode when you reconfigure the ACS.

To reconfigure the ACS:

1. Enter:

```
config acs acs_id
```

The old and new configurations are displayed.

2. Confirm the new configuration.

After confirmation, the database is updated.

If the configuration is unchanged, the configuration is displayed without asking for confirmation, and the utility exits. Examples of this are:

- Only drive types and/or serial numbers change
- LSM serial number changes
- Only LSM type changes between 4410 and 9310
- Number of hands, such as SL8500 robots, change.

However, if these changes occur with other changes requiring confirmation, then you must confirm the new configuration. The database is then updated.

The ACS and its components are removed from the database.

## Limitation of config acs

The `config acs` utility will only update drive configurations in a SCSI/fibre-attached library.

It will not update panel or CAP configurations. In order to update them, you must shutdown ACSLS and use `acsss_config`.

## ■ config drives

The `config drives` utility allows you to reconfigure all drives on an existing drive panel. This includes, adding drives, updating drive types and serial numbers for existing drives, and deleting drives that were removed from the database.

Use the `config drives` utility for dynamic changes to drive configurations, which include installation, replacement, or removal of drives on an existing drive panel. Other changes to tape library hardware configurations such as changes in the number and/or location of storage cells, number or size of CAPs, or replacement of a storage cell panel with a drive panel must be made using the `config lsm` or `config acs` utility. Please note:

- The LSM containing the panel with the changed drive configuration must be online or in diagnostic mode.
- On the drive panel being re-configured, all drives must be ready.
- On a SCSI-attached library, all drives must have been ready when the library was last IPLed, and the drives must still be ready.
- When new drives replace existing drives, varying the LSMs or drive types online or running `config drives` automatically updates the drive types and drive serial numbers.
- You should vary drives offline before removing them. This is not a requirement, but it prevents mount failures to drives that were removed.

## Format

`config drive panel_id` or `config drives panel_id`

## Usage

To reconfigure all drives on an existing drive panel:

1. Enter:

`config drive panel_id` or `config drives panel_id`

The old and new drive configurations are displayed for the panel.

2. Confirm the configuration change.

After confirmation the database is updated.

- If the configuration has not changed, the configuration is displayed without asking for confirmation, and the utility exits.
- If only drive types or serial numbers changed, the ACSLS database is updated without requesting confirmation.

## ■ config lsm

This utility allows you to reconfigure an existing LSM and all of its components. These components include CAPs and panels.

If you want to add or delete an LSM in an ACS, you must use the `config acs` utility.

### Panels:

- If there are panels that are being removed or changed and have been emptied of cartridges, the LSM can remain online.
- If there are panels that are being removed or changed and contain cartridges, it is recommended that you vary the affected LSM to diagnostic until you re-configure the LSM and audit the panels affected.
- If you have added panels and have manually placed cartridges in these panels, please run an audit to reconcile the database.

### Format

```
config lsm lsm_id
```

### Usage

To reconfigure the LSM:

1. Enter:

```
config lsm lsm_id
```

The old and new configurations are displayed.

A “y” next to the panel notifies you that the panel type(s) have changed. Look in the `acsss_config.log` file for details.

2. Confirm the new configuration.

After confirmation, the database is updated.

If the configuration is unchanged, the configuration is displayed without asking for confirmation, and the utility exits.

Minor changes are made automatically without confirmation. Examples are:

- Only drive types and/or serial numbers change
- LSM serial number changes
- Only LSM type changes between the 4410 and 9310
- Number of hands, such as SL8500 robots, change.

However, if these changes occur with other changes requiring confirmation, then you must confirm the new configuration. The database is then updated.

## Limitation of config lsm

The `config lsm` utility will only update drive configurations in a SCSI/fibre-attached library.

It will not update panel or CAP configurations. In order to update them, you must shutdown ACSLS and use `acsss_config`.

## ■ config ports

The `config ports` utility allows you to dynamically add port connections to an ACS.

**CAUTION:** All new ports must be connected to the same ACS as existing ports.

run `config acs acs_id` and then `config ports acs_id`.

**Important:** If you want to replace one ACS with another ACS, you must shutdown ACSLS and use `acsss_config`.

### Format

`config ports acs_id` or `config port acs_id`

### Usage

To add ports:

1. Enter:

`config port acs_id` or `config ports acs_id`

The current port connection for the specified ACS are displayed.

2. Enter the number of port connections for the specified ACS.
3. Specify the device or host ID (s).

**Note:** Ensure that new ports are connected to the same ACS as existing ports.

The old and new configurations are displayed.

A change in the order of the ports is not a configuration change. The connections are displayed on the order they are currently recorded in the database.

4. Confirm the new configuration.

After confirmation, the database is updated.

If the configuration is unchanged, the configuration is displayed, and the utility exits.

## Limitations of config ports

The `config ports acs_id` utility is not supported on a SCSI-attached library.

The `config ports` utility will not delete or change a port (connection) to a library. In order to do this, you must shutdown ACSLS and use `acsss_config`.

## ■ db\_command

The `db_command` command starts or shuts down the database.

**CAUTION:** You must terminate ACSLS using the `idle` command and the `kill.acsss` utility *before* you stop the database.

### Format

```
db_command [start|stop]
```

### Options

`start`  
starts the database.

`stop`  
shuts down the database.

## Usage

- To start the database:  
`db_command start`
- To shut down the database:  
`db_command stop`

## ■ db\_export.sh

The `db_export.sh` utility exports the ACSLS database table data and miscellaneous files in preparation for an upgrade installation or a reinstallation of ACSLS.

### Format

```
db_export.sh [-f db_file]
```

### Options

`-f db_file`

specifies a UNIX file to contain a backup of the ACSLS database. You must have write permissions to the file.

**Note:** If you export the database to a file, the file must reside in a non-volatile directory. If your intention is to re-install ACSLS, the re-installation will destroy the `$ACS_HOME` directory. Consequently, you should place the exported file elsewhere in your file system.

### Usage

Use the `db_export.sh` utility to prepare for a reinstallation of ACSLS or an upgrade installation of ACSLS.

### Examples

Refer to [Chapter 11, “Database Backup and Restore”](#) for examples.

See also:

- [“db\\_import.sh”](#)
- [“rdb.acsss” on page 275](#)
- [“Exporting the Database” on page 206](#)

## ■ db\_import.sh

The `db_import.sh` utility imports the ACSLS database table data and the policy files that you exported when you used the `db_export.sh` utility. These miscellaneous files are located in `$ACS_HOME/data/external` and define your policies for ACSLS. They specify Access Control settings, scratch preferences, Extended Store LSMs, custom `volrpt` settings, volume attributes (for `watch_vols` utility), etc. The `db_import.sh` utility also provides disaster recovery capabilities, as well as the retention of your customized dynamic variables when moving to a different operating system or from an earlier release.

### Format

```
db_import.sh [-f db_file]
```

### Options

`-f db_file`  
specifies a UNIX file created by `db_export.sh`.

### Usage

Use the `db_import.sh` utility to import the ACSLS database that you exported using the `db_export.sh` utility.

**CAUTION:** The `db_import` utility will not run if ACSLS is running.

See also

- [“db\\_export.sh” on page 253](#)
- [“rdb.acsss” on page 275](#)
- [“Importing the Database” on page 209](#)

## ■ del\_vol

The `del_vol` utility deletes from the database volumes that either:

- cannot be found in the library, or
- are marked as absent or ejected, or
- are in an offline LSM or drive (missing).

**Note:** This utility does not delete a volume that is found in the library.

You can use the `del_vol` utility to remove a volume record without waiting for the expiration of an absent or ejected status.

## Format

```
del_vol [-n] [-d] [-q] vol_id
```

## Options

-n

Optional; no-confirm mode; delete volume without prompting user if all locations can not be examined.

-q

Optional; quiet mode; do not print out all information extracted from the database.

-d

Optional; delete the volume; do not mark it absent. Use this option to remove absent or ejected volumes from the database.

*vol\_id*

The volume serial number to be deleted.

The Manual Volume Delete (`del_vol`) utility can now retain volumes as absent. Using the `-d` option deletes the specified volume without waiting for the expiration of an absent or ejected status.

**Note:** If all of the referenced locations for the volume can't be verified (i.e., the LSM is offline or the drive is not ready), you are prompted to confirm the deletion unless the `-n` (`no_confirm_flag`) is on. After confirmation, or if the `no_confirm_flag` is on, the volume and information associated with it, such as scratch pool membership and current and pending locks, are removed from the database. Volume-related information is displayed unless the `-q` (`quiet_flag`) has been entered. If multiple options are used, they can be formatted either as separate options or as a contiguous string.

## Usage

ACSLs and the database must be up and running (*not* idle) to use this utility. Typically, you use the `del_vol` utility to delete a volume from an offline LSM so that you can re-enter the cartridge into the CAP of an online LSM. Make sure that you manually remove the same volume as the volume you deleted from the database. If you delete a volume from the database, but mistakenly remove another cartridge from the LSM, return the cartridge to its proper cell in the LSM and then remove the correct cartridge.

You can use `del_vol` to remove a cartridge from an offline LSM and then reenter it in an online LSM, so it can be automatically mounted. To do this:

- Remove the cartridge from the offline LSM.
- Use `del_vol` to mark the cartridge as absent.

- Enter the cartridge into the online LSM.

Using `del_vol`, removes from the database the cartridge and all information associated with it, such as scratch pool membership and current and pending locks (provided the absent volume retention period is zero). All available information related to the cartridge is printed out and verified. You must manually remove the cartridge from the LSM.

**CAUTION:** If you mistakenly delete a cartridge from the database, you should audit the sub-panel containing the home cell of the deleted cartridge to re-enter the cartridge into the database.

Running `del_vol` while the system is in recovery can produce unpredictable results. The recovery sequence happens during a vary LSM online.

## Example

To delete cartridge U01102 without a printout of the cartridge information:

```
del_vol -q U01102
```

## See Also

- [“Using the Manual Volume Delete Utility” on page 181](#)

## ■ drives\_media.sh

This routine displays all drive types, media types and drive-to-media compatibility that are supported by the current release of ACSLS. The information is normally displayed to standard output.

### Format

```
drives_media.sh [ -f, -h ]
```

### Options

- f  
the information is written to three files:
  - /tmp/drive\_types.txt
  - /tmp/media\_types.txt
  - /tmp/media\_compatibility.txt.
- h  
displays the syntax message.

### Example

```
$ drives_media.sh
```

```

#
# Tape Media Supported by ACSLS
#
# ACSLS uses "Media Name" when reporting cartridge media type in
# cmd_proc, display and event log messages.
#
# "Library Media Domain" and "Library Media Type" are the values passed
# from the library hardware to ACSLS.
#
# The "ACSLs Media Type Number" is used in the ACSAPI to report the media
# type to ACSAPI clients. (This number is also used internally in ACSLS.)
#
# In the "Cleaning Cartridge?" field, three values may be defined:
# Yes - This media type is used exclusively for cleaning.
# No - This media type is never used for cleaning.
# Maybe - This media type may apply to a data or a cleaning cartridge.
#
# Media type 24 is reserved for VSM virtual tape media.
# Virtual tapes are supported by LibraryStation, which uses media type
# 24 for virtual media in its ACSAPI communication with clients.
# To avoid conflicts, ACSLS must NOT use media type 24.
#

```

Media Type Name	Media Domain	Library Media Type	ACSLs Media Type Number	Cleaning Cartridge?	Max Cleaning Usage
3480	00	1	0	Maybe	100
3490E	00	E	1	No	0
DD3A	00	A	2	No	0
DD3B	00	B	3	No	0
DD3C	00	C	4	No	0
DD3D	00	D	5	Yes	50
DLTIII	01	C	6	Maybe	20
DLTIIIIXT	01	E	8	No	0
DLTIV	01	D	7	No	0
EECART	00	Z	11	No	0
JLABEL	00	J	12	Maybe	100
KLABEL	00	K	15	Maybe	100
LTO-100G	4C	1	16	No	0
LTO-10GB	4C	C	19	No	0
LTO-200G	4C	2	26	No	0
LTO-35GB	4C	B	18	No	0
LTO-50GB	4C	A	17	No	0
LTO-CLN1	43	1	22	Yes	15
LTO-CLN2	43	2	20	Yes	50
LTO-CLN3	43	3	21	Yes	50
LTO-CLNU	43	U	25	Yes	50
SDLT	01	S	23	Maybe	20
SDLT-2	01	2	27	No	0
STK1R	00	R	9	No	0
STK1U	00	U	10	Yes	100
STK2P	00	P	13	No	0
STK2W	00	W	14	Yes	100
VIRTUAL	00	v	24	No	0

```

# Drive and Media Compatibility
#
# This shows the media types that are compatible with each drive type.
#

```

Drive Name	Compatible Media Name
4480	3480
4490	3480
4490	3490E
4490	JLABEL
4490	KLABEL
9490	3480

## Utility Reference

9490	3490E
9490	JLABEL
9490	KLABEL
SD3	DD3A
SD3	DD3B
SD3	DD3C
SD3	DD3D
SD3	JLABEL
SD3	KLABEL
4890	3480
4890	3490E
DLT2000	DLTIII
DLT2000XT	DLTIII
DLT2000XT	DLTIIIIXT
DLT4000	DLTIII
DLT4000	DLTIIIIXT
DLT4000	DLTIV
DLT7000	DLTIII
DLT7000	DLTIIIIXT
DLT7000	DLTIV
9840	STK1R
9840	STK1U
9840-3590	STK1R
9840-3590	STK1U
9491	EECART
9491	3490E
9491	3480
9491	JLABEL
9491	KLABEL
DLT8000	DLTIII
DLT8000	DLTIIIIXT
DLT8000	DLTIV
T9940A	STK2P
T9940A	STK2W
9940-3590	STK2P
9940-3590	STK2W
T9840B	STK1R
T9840B	STK1U
T9840B35	STK1R
T9840B35	STK1U
HP-LTO	LTO-100G
HP-LTO	LTO-50GB
HP-LTO	LTO-35GB
HP-LTO	LTO-10GB
HP-LTO	LTO-CLN1
HP-LTO	LTO-CLNU
IBM-LTO	LTO-100G
IBM-LTO	LTO-50GB
IBM-LTO	LTO-35GB
IBM-LTO	LTO-10GB
IBM-LTO	LTO-CLN2
IBM-LTO	LTO-CLNU
SGT-LTO	LTO-100G
SGT-LTO	LTO-50GB
SGT-LTO	LTO-35GB
SGT-LTO	LTO-10GB
SGT-LTO	LTO-CLN3
SGT-LTO	LTO-CLNU
SDLT	SDLT
SDLT	DLTIV
SDLT-320	SDLT
SDLT-320	DLTIV
T9940B	STK2P
T9940B	STK2W
T9940B35	STK2P
T9940B35	STK2W
VIRT-DRV	VIRTUAL
T9840C	STK1R
T9840C	STK1U
T9840C35	STK1R

```

T9840C35          STK1U
HP-LTO-2          LTO-200G
HP-LTO-2          LTO-100G
HP-LTO-2          LTO-50GB
HP-LTO-2          LTO-35GB
HP-LTO-2          LTO-10GB
HP-LTO-2          LTO-CLN1
HP-LTO-2          LTO-CLNU
IBM-LTO-2         LTO-200G
IBM-LTO-2         LTO-100G
IBM-LTO-2         LTO-50GB
IBM-LTO-2         LTO-35GB
IBM-LTO-2         LTO-10GB
IBM-LTO-2         LTO-CLN2
IBM-LTO-2         LTO-CLNU
SGT-LTO-2         LTO-200G
SGT-LTO-2         LTO-100G
SGT-LTO-2         LTO-50GB
SGT-LTO-2         LTO-35GB
SGT-LTO-2         LTO-10GB
SGT-LTO-2         LTO-CLN3
SGT-LTO-2         LTO-CLNU
SDLT-600          SDLT
SDLT-600          SDLT-2

```

```

# Tape Drives Supported by ACSLS
#
# "Drive Name" is used to report the drive type in cmd_proc, display and
# and event log messages.
#
# "ACSLs Drive Number" specifies the drive type in ACSAPI communications
# with clients. (It is also used internally by ACSLS to identify the
# drive type.)
#
# Note: Drive type 23 is reserved for VSM virtual drives
#       (not supported by ACSLS).
#

```

Drive Name	Drive Description	ACSLs Drive Type Number
-----	-----	-----
4480	18 track	0
4490	Silverton 36 track	1
9490	Timberline 36 track High Perf	2
SD3	Redwood helical	3
4890	Twin Peaks - 36 track	4
DLT2000	Quantum DLT2000	5
DLT2000XT	Quantum DLT2000XT	6
DLT4000	Quantum DLT4000	7
DLT7000	Quantum DLT7000	8
9840	9840	9
9491	Timberline EE - 36 track	10
DLT8000	Quantum DLT8000	11
9840-3590	IBM 3590 emulation (9840)	12
T9940A	T9940A with 3490E image	13
9940-3590	T9940A with 3590 image	14
SDLT	Super DLT 220	15
T9840B	High Perf 9840 with 3490E image	16
T9840B35	T9840B with 3590 image	17
HP-LTO	HP LTO	18
IBM-LTO	IBM LTO	19
SGT-LTO	Seagate LTO	20
T9940B	T9940B with SCSI or 3490E image	21
T9940B35	T9940B with 3590 image	22
VIRT-DRV	VSM virtual drive	23
SDLT-320	Super DLT 320	24
T9840C	T9840C with Native or 3490E image	25
T9840C35	T9840C with 3590 image	26
HP-LTO-2	HP LTO Generation 2	27

IBM-LTO-2	IBM LTO Generation 2	28
SGT-LTO-2	Seagate LTO Generation 2	29
SDLT-600	SDLT-600	30

\$

## ■ ejecting.sh

The `ejecting.sh` utility takes a list of cartridges intended for ejecting and creates a file containing a set of eject commands optimized for the specified CAP. The input file must contain a simple list of cartridges that are to be ejected. The resulting series of optimized eject commands are sent to standard out and can be piped directly to `cmd_proc`

### Format

```
ejecting.sh <CAP-id> <volume list>
```

### Options

- CAP-id  
the CAP containing the cartridges to be ejected.
- volume list  
list of cartridges to be ejected.

### Usage

- This script checks for the validity of the CAP-id argument and if the specified CAP does not exist, it displays the message "CAP <CAP-id> not found".
- The script also determines the number of cells in the specified CAP and then formats a series of eject commands to completely fill that cap. This is repeated until all of the cartridges in the incoming cartridge list have been included in the output.

### Examples

#### Example 1:

In this example a series of optimized eject commands are sent to standard out and piped directly to `cmd_proc`:

```
ejecting.sh 0,0,0 volume_list_file | cmd_proc -lq
```

#### Example 2:

The output is redirected to a file by one of the following methods:

```
ejecting.sh 0,0,0 volume_list_file > eject.dat
```

```
ejecting.sh 0,0,0 volume_list_file | tee eject.dat
```

**Note:** If the output of this script has been redirected to a file, you may use that file as input for `cmd_proc` as follows:

```
cmd_proc -lq < eject.dat
```

## ■ find\_tty.sh

The `find_tty.sh` utility identifies available ports that can be used by ACSLS user, 'acsss'. This utility reviews all defined ports and verifies which ports are actually available. All available ports are displayed and the output reveals which of these have read-write permissions for use by 'acsss'. In addition, other available ports which are not currently accessible to user 'acsss' are displayed.

The method to identify available ports varies from one system to the next. For Solaris, the device nodes for all ports defined in `/dev/term` are displayed. For AIX, the method is a "lsdev" query.

### Format

```
find_tty.sh
```

### Options

none

### Usage

- This option allows you to select an appropriate device on your system. You select a device when following prompt is displayed:

```
Connect the LMU to an appropriate serial device
```

- If you do not know which device to use, run `find_tty.sh` to identify all eligible devices.

## Example

The following example is of an output from a Solaris box with multiple ports.

```
$ find_tty.sh
/dev/term/0 is accessible to acsss
/dev/term/1 is accessible to acsss
/dev/term/2 is accessible to acsss
/dev/term/3 is accessible to acsss
/dev/term/4 is accessible to acsss
/dev/term/5 is accessible to acsss
/dev/term/6 is accessible to acsss

/dev/term/0 is not accessible to acsss.
You must set permissions on the device node:
crwx----- root /devices/sbus@1f,0/SUNW, spif@1,0:7
/dev/term/a is accessible to acsss
/dev/term/b is accessible to acsss
```

## ■ free\_cells.sh

The `free_cells.sh` utility allows you to monitor and manage the free cells in libraries managed by ACSLS. This utility reports the free cell count for LSM, ACS, and the ACSLS server.

**Note:** Since `free_cells.sh` reads every cell record in the database, it is recommended that you run this utility infrequently, such as only during periods of low database activity.

This utility is located in the `$ACS_HOME/diag/bin` directory

### Format

```
free_cells.sh
```

### Option

`-a`  
displays the free cells, allocated cells and the total number of cells in each ACS and LSM, as well as those managed by the ACSLS server.

## Examples

### **Free cells in each LSM**

- LSM is not a L5500 (lists only the total free cells)

- No options selected

```
LSM 1,3
    Total free cells      = 2,345
```

- -a option (also lists allocated cells and total cell count)

```
LSM 1,3
    Total free cells      = 3,345
    Allocated cells       = 3,155
    Total cells           = 6,500
```

- LSM is a L5500 (lists LTO and non-LTO free cells separately)

- No options selected

```
LSM 0,2
    LTO free cells        = 573
    non-LTO free cells    = 467
    Total free cells      = 1,040
```

- -a option (also lists allocated cells and total cell count)

```
LSM 0,2
    LTO free cells        = 573
    non-LTO free cells    = 467
    Total free cells      = 1,040
    Allocated cells       = 4,460
    Total cells           = 5,500
```

### **Free cells in each ACS**

If there is only one LSM in the ACS, only the ACS is listed as shown in the following examples.

- ACS does not include any L5500s

- No options selected (displays only the total free cell count)

```
ACS 1
    Total free cells      = 5,342
```

- -a option (also displays allocated cells and total cell count)

ACS 1

```
Total free cells      = 5,342
Allocated cells       = 5,658
Total cells           = 11,000
```

- ACS includes L5500 (displays LTO and non-LTO free cells separately)

- No options selected

ACS 0

```
LTO free cells        = 1,573
non-LTO free cells    = 968
Total free cells      = 2,541
```

- -a option (also displays allocated cells and total cell count)

ACS 0

```
LTO free cells        = 1,573
non-LTO free cells    = 968
Total free cells      = 2,541
Allocated cells       = 2,959
Total cells           = 5,500
```

***Free cells managed by the ACSLS server***

- ACS does not include any L5500s

In this case only the total free cell count is displayed

- No options selected

ACSL Server

```
Total free cells      = 7,883
```

- -a option

ACSL Server

```
Total free cells      = 7,883
Allocated cells       =14,117
Total cells           =22,000
```

- ACS includes L5500s

In the case where there are L5500 libraries being managed, only the LTO free cell and total free cell count is displayed. Non-LTO free cells are not displayed since the L5500 library is the only library that separates LTO from non-LTO cells.

- No options selected

ACSLs Server

LTO free cells	= 1,573
Total free cells	= 7,883

- -a option

ACSLs Server

LTO free cells	= 1,573
Total free cells	= 7,883
Allocated cells	=14,117
Total cells	=22,000

## ■ get\_license\_info

The `get_license_info` utility displays the following information about the license:

- Customer
- Site
- Product
- Key type
- Expiration date
- Days before license expires
- Licensed cells (volumes)
- License key
- Used cells (volumes)
- Free cells (volumes)
- Total cells (volumes)

### Format

`get_license_info`

### Options

none

## Example

```
$ get_license_info
Customer      STK_HQ_SUPPORT
Site          6111
Product       ACSLS700
Key Type      Permanent
Expires       2010-12-31
Days to Expire 2430
Licensed Cells 600
License Key   WIRLJP7F2YBIVIO8KMS
Used Cells    7
Free Cells    425
Total Cells   432
```

## ■ greplog

Use the `greplog` utility to filter the `acsss_event` log to include or to exclude messages containing specific keywords. The syntax of this routine is similar to the Unix 'grep' function. `greplog` is specifically designed for use with the `acsss_event.log`, but it may function with any type of message file where the records are separated by an empty line.

### Format

```
greplog -[v|i] <keyword> <logfile>
```

### Options

- v  
Optional. `greplog` displays all the messages in the log except those which include the keyword.
- i  
Optional. `greplog` ignores the characters in the specified keyword.
- keyword  
returns the complete multi-line message containing the keyword.
- logfile  
list of log files.

### Usage

Since the utility is specifically designed for log files, `greplog` returns the complete multi-line message containing the keyword rather than a single line

containing that word. Using the `-i` option, `greplog` ignores the characters in a specified keyword. Using the `-v` option, `greplog` displays all of the messages in the log except those which include the keyword. `greplog` is specifically designed for use with the `acsss_event.log`, but it may function with any type of message file where the records are separated by an empty line.

## ■ kill.acsss

The `kill.acsss` utility terminates ACSLS.

### Format

```
kill.acsss
```

### Options

None.

### Usage

Use the `kill.acsss` utility to terminate ACSLS.

### See also

- [“rc.acsss” on page 274](#)
- [“idle” on page 322](#)

**CAUTION:** *Do not run `kill.acsss` while an ACSLS request is in progress! You must first idle ACSLS (with the `idle` command) before you run `kill.acsss`.*

**Note:** `kill.acsss` does *not* shut down the database. You must issue `db_command stop` if you want to shut down the database after you shut down ACSLS.

## ■ moving.sh

The `moving.sh` utility moves multiple cartridges to one or more LSMs. This utility reads a file that lists the cartridges to be moved. These cartridges can be:

- Cartridges in one or more LSMs
- Cartridges on a panel being moved to other panels in the same LSM or other LSMs
- Any group of cartridges you select

The limitations of *moving.sh* are:

- All destination LSMs and cartridges in the *vol\_list\_file* must be in same ACS
- If any destination LSM is offline or does not contain any free cells, no cartridges are moved to that LSM

**Notes:**

- The *moving.sh* utility runs only if ACSLS is running
- Internally, *moving.sh* moves only one cartridge at a time to avoid impacting library performance (mounts and dismounts)
- You can run multiple move utilities in parallel after creating separate lists of volumes. Please note that if:
  - the destination LSM is same, you should make sure that there are enough free cells in the LSM to accommodate all cartridges
  - you are moving within one SL8500 library - there are only two elevators, so performing more than two move utilities will not increase performance

## Format

```
moving.sh -f vol_list_file -t lsm_id...
```

Where:

- *-f vol\_list\_file*

The file containing the list of volumes to be moved

- *-t lsm\_id*

Specifies one or more LSM IDs to which the cartridges will be moved. Each LSM ID should be separated by a space

## Usage

Use the *moving.sh* utility to move a list of cartridges to other LSM(s) or from one panel to another panel in the same LSM or in another LSM.

You can use either a custom volume report or the *display* command to create a file containing the list of volumes to be moved from an LSM.

You want to use the *moving.sh* utility:

- When a SL8500 is initially partitioned or repartitioned, and one or more rails (LSMs) are removed from an existing partition (ACS), *moving.sh* can move cartridges from the LSM being removed from the partition to the LSM(s) that will remain in the partition

- When any LSM(s) is removed from an ACS, *moving.sh* can move cartridges to the LSMs that will remain in the ACS.

For example, if SL8500s are removed from a library complex (ACS), *moving.sh* moves cartridges from the SL8500s that are being removed to the LSMs that will remain in the library. This also applies when 9310 LSMs are removed from an ACS of 9310s.

- When a storage expansion module(s) is removed from an SL8500, cartridges can be moved from the panels being removed to the panels that will remain in the library
- To optimize library performance you may need to move inactive cartridges to an LSM with few or no drives that are used to archive cartridges. This frees up space in LSMs with drives for new, active cartridges.

### **Creating the Volume List File**

You can use either the *volrpt* (custom volume report) or the *display* command to create a file containing the list of volumes to be moved from an LSM. Before you begin, you must create the file containing the list of volumes to be moved from an LSM using either *volrpt* or the *display* command:

- Using *volrpt* to create *vol\_list\_file* -

```
volrpt -d -f custom_volrpt_input -l 0,4 > vol_list_file
```

Where the custom-*volrpt\_input* file is:

```
VOLUME_ID 6
```

- Using *display* command to create *vol\_list\_file* -

```
display volume * -home acs,lsm,panel,*,* -f volume
```

This selects all volumes on the panel identified by the *-home* parameter. Row and column are wild-carded. Only the *vol\_id* is output. This output should then be copied to a *vol\_list\_file*.

### **Procedures for Moving a Group of Cartridges**

The following procedures describe how to:

- move cartridges before removing an LSM from an ACS
- move cartridges before changing or removing panels

### **Moving Cartridges Before Removing an LSM from an ACS**

After a library is re-configured or re-partitioned and if an LSM is removed from an ACS, all cartridges in the LSM become inaccessible. So, before the LSM is removed, all its cartridges should be moved to LSM(s) that will remain in the ACS. Use the following procedure:

- When a rail (LSM) is removed from a partition in a partitioned SL8500.
- When an LSM(s) is removed from an ACS. The ACS can include 9310s or an SL8500 library.

1. Plan your new configuration.

- Organize the cartridges and drives for performance.
- Empty an LSM shortly before you change the library configuration.
- Determine how many cartridges you have in the LSMs that you are emptying, and how many free cells in the LSMs to which you are moving cartridges.

Use `free_cells.sh -a` to find out the number of cartridges in these LSMs (allocated cells) and free cells.

2. Schedule the move and reconfiguration.

- Schedule the move to minimize the impact on your system.

Moving the cartridges takes time, and reconfiguring a library or repartitioning an SL8500 is disruptive.

- Make sure there are enough free cells in the target LSM(s) for the cartridges being moved. If you have to, eject cartridges to free up space.

3. Vary all of the drives in the LSM being removed offline.

This prevents:

- Contention for robots in the LSM.
- Mounts to this LSM.

Otherwise, cartridges mounted to this LSM can float to new home cells in the LSM, filling up the LSM that you are trying to empty.

4. Vary the LSM being emptied to diagnostic mode to restrict access to only the `cmd_proc` using:

```
vary lsm lsm_id diagnostic
```

Example: `vary lsm 0,1 diagnostic`

5. Run a custom `volrpt` to output all of the cartridges in the LSM being emptied to a file, using the following command:

```
volrpt -f custom_volrpt_input -l from_lsm_id > move_vols_list
```

Where the `custom_volrpt_input` is:

```
VOLUME_ID 6
```

Example: `volrpt -f volrpt_input -l 0,1 > move_vols_list`

Refer to [“Creating a Custom Volume Report” on page 233](#) for more information.

6. Move the cartridges out of the LSM being emptied, using the following:

```
moving.sh -f move_vols_list -t dest_lsm_id(s)
```

7. Check that the LSM is empty using `volrpt` since cartridges may have been entered into the LSM or may have “floated” into it.

```
volrpt -l from_lsm_id
```

If it is not empty run the custom `volrpt` again to select the volumes that are now in the LSM. Then, run *moving.sh* again (steps 3 and 4).

**Note:** Do not run *moving.sh* again with the original list of volumes.

8. Vary the LSM being emptied offline to prevent volumes from being moved to it.

```
vary lsm lsm_id offline
```

9. Remove the LSM from the partition and/or ACS.
10. Reconfigure the ACS, using either `config acs acs_id` or `acsss_config`.

### ***Moving Cartridges Before Changing or Removing Panels***

You need to move the cartridges before changing a cell panel to a drive panel in a 9310 or removing a storage expansion module in an SL8500.

1. Same procedure as above.
2. Same procedure as above.
3. Same procedure as above.
4. Same procedure as above.
5. Select the cartridges in the panel being emptied, and output them to a file.
  - a. Run a custom `volrpt` to output all of the cartridges in the LSM being emptied to a file. Include the panel number (in the home cell ID).

```
volrpt -f custom_volrpt_input -l from_lsm_id > move_vols_list_1
```

Where the *custom-volrpt\_input* is:

```
VOLUME_ID 6
CELL_ID 13
```

Select the volumes in the panel(s) being emptied and output these `vol_ids` to your *move\_vols\_list\_2*.

- b. Select the cartridges in a panel being emptied using the `display volume` command.

```
display volume * -home acs,lsm,panel,*,* -f volume > move_vols_list_2
```

This selects all volumes on the panel identified by the *-home* parameter. The row and column are wild-carded. Only the *vol\_id* is output, and the output is written to the file.

**Note:** If the destination or “to” LSM is the same as the source or “from” LSM and more than one panel is being emptied, some of the volumes will be moved back to the panels being emptied. You will have to select the volumes off the panels and move them repeatedly to clear out the panels.

6. Same procedure as above.
7. Same procedure as above.
8. Same procedure as above.
9. Same procedure as above.
10. Reconfigure the LSM, using either `config lsm lsm_id` or `acsss_config`.

## Examples

- Moving cartridges from LSM 0,4 to LSM 0,0 and 0,1

To move cartridges from LSM 0,4 to LSM 0,0 and 0,1, you must first create a file containing the list of cartridges in LSM 0,4 using *volrpt*, and then run the *moving.sh* utility as below:

### Sample output

```
$ moving.sh -f vol_list.txt -t 0,0 0,1
Number of free cells in LSM 0,0 : 308
Number of free cells in LSM 0,1 : 362
-----
Total number of free cells : 670
Total number of volumes to move : 7
Cartridge CAB001 moved to 0,0,3,0,0
Cartridge CAB002 moved to 0,0,4,0,0
Cartridge CAB003 moved to 0,0,5,0,0
Cartridge CAB004 moved to 0,0,6,0,0
Cartridge CAB005 moved to 0,0,7,0,0
Cartridge CAB006 moved to 0,0,8,0,0
Cartridge CAB007 moved to 0,0,9,0,0

Summary
=====
Number of free cells remaining in LSM 0,0 : 301
Number of free cells remaining in LSM 0,1 : 362
-----
Total number of free cells remaining : 663
Number of cartridges moved : 7
Number of cartridges not moved : 0
```

- Moving cartridges from LSMs 0,4 0,5 0,6 and 0,7 to LSMs 0,0 0,1 0,2 and 0,3,

To optimize performance by moving each LSM to the adjacent LSM:

1. Prepare files containing the list of cartridges in LSM 0,4 0,5 0,6 and 0,7 using *volrpt*.
2. Run four *moving.sh* utilities at the same time but in separate Unix command terminals.

There is no contention between the separate instances of *moving.sh* because the source and destination LSMs and the pass-thru ports used are all different:

### Sample output

```
moving.sh -f vol_list_0-4.txt -t 0,0
moving.sh -f vol_list_0-5.txt -t 0,1
moving.sh -f vol_list_0-6.txt -t 0,2
moving.sh -f vol_list_0-7.txt -t 0,3
```

## Managing Cartridges for Performance

The *moving.sh* utility can be used to move inactive cartridges to archival LSMs. An archival LSM is an LSM with few or no drives that is used to store cartridges that have a low probability of being mounted. The top rail in an SL8500 is a good choice for an archival LSM because it does not have direct access to the CAP.

Inactive cartridges that do not need to be in a library can be ejected, while inactive cartridges that still need to be available for automated mounts should be moved to archival LSMs.

To move inactive cartridges to an archival LSM:

1. Identify the inactive cartridges. For example to select cartridges that have not been accessed in the last three months:
2. Run a custom *volrpt* to output all of the cartridges in the LSM being examined, and output the results to a file.

```
volrpt -f custom_volrpt_input -l from_lsm_id > move_vols_list_1
```

Where the *custom-volrpt\_input* is:

```
VOLUME_ID 6
ACCESS_DATE 15
```

3. Select the cartridges where the `access_date` is earlier than three months ago and output these `vol_ids` to a file with the list of volumes to be moved.
4. Move the inactive cartridges to the archival LSM.

```
moving.sh move_vols_list_2 archival_lsm_id
```

### See also

- [“display volume” on page 409](#)
- [“volrpt” on page 281](#)

## ■ probeFibre.sh

This utility displays the model number, revision level, and Target-LUN address of each device connected behind an Emulex (LP10000) or QLogic (QLA2300) fibre-channel HBA. It uses the Emulex utility 'hbacmd' and the QLogic utility 'scli' to display the library devices that are visible through the respective Emulex or QLogic fibre HBA. The model number of each library device along with its target and LUN ids can be displayed using the `-v` option.

The `probeFibre.sh` utility can be run even before the `mchanger` devices are created for each library.

### Format

```
probeFibre.sh
```

## ■ rc.acsss

The `rc.acsss` utility brings ACSLS up and automatically starts the database. There is no need to manually start the database.

### Format

```
rc.acsss [idle]
```

### Options

`idle` brings ACSLS up in the idle state.

### Usage

Use the `rc.acsss` utility to start ACSLS. When you start ACSLS, it goes through recovery mode to online state. ACSLS initialization includes checking its database and “recovering” status of the library cartridges and library hardware.

## Examples

```
rc.acsss
```

## See Also

- [“kill.acsss” on page 267](#)
- [“start” on page 376](#)

**Note:** If you are not logged in as `acsss` and try to run `rc.acsss`, the system prompts you for the `acsss` password.

You can run only one copy of ACSLS on the server. `rc.acsss` will not restart ACSLS if it is already running.

**Note:** `rc.acsss` automatically starts the database when it starts ACSLS.

## ■ rdb.acsss

The `rdb.acsss` utility restores the ACSLS database and the policy files using a backup created by either the automatic backup function or the `bdb.acsss` utility. These miscellaneous files are located in `$ACS_HOME/data/external` and define your policies for ACSLS. They specify Access Control settings, scratch preferences, Extended Store LSMs, custom `volrpt` settings, volume attributes (for `watch_vols` utility), etc.

## Format

```
rdb.acsss
```

**Note:** Do not specify the `-f` option when you run the `rdb.acsss` utility.

If you backed up your database to an external network file or to an alternate tape device, you use the `-f` option only after entering `rdb.acsss` and then choosing the third recovery option. When prompted, enter `-f` and the path name to your external network file or alternate tape device.

## Menu Options

When you run `rdb.acsss`, a menu displays five options:

**Figure 43. rdb.acsss Menu Options**

```

Please enter the number followed by Return for your choice from
the following menu.

Press? followed by the Return key for help.

1: Restore from current local disk backup
2: Restore from previous local disk backup (to a point in time)
3: Restore from a previous tape or network file backup
4: Restore a backup created on a different server
5: Restore ACSLS non-database files
E: Exit

```

1. Restore from current local disk backup

**Explanation:** The database is recovered from the most recent backup to disk, and the transaction logs are applied to restore all committed updates. Miscellaneous ACSLS files are restored.

**Usage:** Use this option to restore a corrupted or lost database or to recover the database on the primary disk from a backup on the secondary disk.

2. Restore from previous local disk backup (to a point in time)

**Explanation:** The database is restored to the last backup before the specified date and time within the backup retention period. Archived transaction logs are applied up to the specified date and time. Miscellaneous ACSLS files are restored from the backup only.

**Usage:** Use this option to restore the database back to a specific date and time.

3. Restore from a previous tape or network file backup

**Explanation:** The database is restored from an external, manual backup created by `bdb.acsss`. The backup is on either tape or an external network file. Miscellaneous ACSLS files are restored. Logical logs are not applied.

**Usage:** Use this option to recover the ACSLS environment after a server failure when both disks are corrupted or lost.

#### 4. Restore ACSLS from a server-to-server backup

**Explanation:** The database is restored from a server-to-server backup created by `bdb.acsss` using the `-s` option.

**Usage:** Use this option to recover ACSLS to a standby server. In this configuration, ACSLS runs on a primary server. If the primary server fails, the most recent backup is restored to a redundant standby server.

**Note:** The backup must have been created with the `bdb.acsss -s` option.

#### 5. Restore ACSLS non-database files

**Explanation:** Only the non-database, miscellaneous ACSLS files are restored from a manual tape backup or an external network file backup.

**Usage:** Use this option to restore ACSLS data existing in `data/external:` access control files, custom `volrpt` files, fixed volume file, and scratch preferences file.

#### 6. Exit

Exits the `rdb.acsss` utility. When you exit the `rdb.acsss` utility, you are prompted to do a backup. If you choose not to do a backup, ACSLS automatically performs a backup to a local disk.

## See Also

Recovery procedures for:

- [“Restoring the Database to the Most Recent Backup” on page 222](#)
- [“Restoring the Database to a Specified Date and Time” on page 223](#)
- [“Recovering from a Primary Disk Failure” on page 225](#)
- [“Recovering from a Second Disk Failure” on page 226](#)
- [“Recovering from a Failed Server” on page 227](#)
- [“Restoring Miscellaneous ACSLS Files” on page 229](#)

- [“Backing up ACSLS and Restoring to Another Server” on page 218](#)
- [“bdb.acsss” on page 243](#)

## ■ showDevs.sh

The `showDevs.sh` utility displays the critical device attributes associated with each `mchanger` instance in the `/dev` directory. Critical attributes include the library model number, cell capacity, number of attached drives, and Target-LUN address. The utility also displays the model number of the host bus adapter that is associated with each `mchanger` device. Additional attributes can be displayed using the following options.

### Format

```
showDevs.sh [-s][-w][-q][-sq][-wq]
```

### Usage

The utility can be run as `root` or as `acsss`. Each user ID provides a different set of capabilities. For example:

- `acsss`  
 user `acsss` can list the serial number and the number of cells and drives associated with each library
- `root`  
`root` can extract serial number and the number of cells and drives only for libraries that are not online to ACSLS.

However, `root` is able to reveal the Wide Node Name and World Wide Port Name of each fibre-attached library.

### Options

This utility can be run with several options.

no argument  
 reveals each HBA and the model, revision level, target and LUN of each attached library.

-s  
 Optional. This option displays the serial number of each library along with the number of cells and the number of attached drives associated with each library.

-w  
 Optional. This option displays the World Wide Names (WWNN and WWPN) for each HBA and for each attached target library.

The "-w" option is not available for user acsss.

- q  
Optional. This option displays only essential mchanger data, while excluding information about the HBAs.
- sq  
Optional.
- wq  
Optional.

## Examples

```
# whoami
root

# pwd
/export/home/ACSSS/diag/bin

# ./showDevs.sh
QLogic QLA2300  QLGC,qla@1
/dev/mchanger0: STK L700 V-1012 Target-3 LUN-0
/dev/mchanger1: STK L700 V-1012 Target-3 LUN-1
/dev/mchanger2: STK VLIBRARY V-1012 Target-4 LUN-1
/dev/mchanger3: STK VLIBRARY V-1012 Target-4 LUN-2

LSILogic 53c1030  LSILogic,scsi@1
/dev/mchanger4: STK SL500 Target-0 LUN-0

Emulex LP10000  fibre-channel@2
/dev/mchanger5: STK L180 Target-0 LUN-0

# ./showDevs.sh -q
/dev/mchanger0: STK V700 216-cells 10-drives
/dev/mchanger1: STK V700 384-cells 10-drives
/dev/mchanger2: STK VLIBRARY 1000-cells 50-drives
/dev/mchanger3: STK VLIBRARY 10000-cells 461-drives
/dev/mchanger4: STK SL500 Target-0 LUN-0
/dev/mchanger5: STK L180 Target-0 LUN-0

# ./showDevs.sh -wq
Target-3 WWNN-50-01-04-F0-00-7E-B3-39 WWPN-50-01-04-F0-00-7E-B3-3A
/dev/mchanger0: STK L700 V-1012 Target-3 LUN-0
/dev/mchanger1: STK L700 V-1012 Target-3 LUN-1

Target-4 WWNN-50-01-04-F0-00-7E-B3-45 WWPN-50-01-04-F0-00-7E-B3-46
/dev/mchanger2: STK VLIBRARY V-1012 Target-4 LUN-1
/dev/mchanger3: STK VLIBRARY V-1012 Target-4 LUN-2

/dev/mchanger4: STK SL500 Target-0 LUN-0

Target-0 WWNN-50:01:04:F0:00:41:18:46 WWPN-50:01:04:F0:00:41:19:32
/dev/mchanger5: STK L180 V-0310 Target-0 LUN-0
```

```

# ./showDevs.sh -w
QLogic QLA2300  QLGC,qla@1
    HBA:  WWNN 20-00-00-E0-8B-07-AA-3A  WWPN 21-00-00-E0-8B-07-AA-3A

    Target-3 WWNN-50-01-04-F0-00-7E-B3-39 WWPN-50-01-04-F0-00-7E-B3-3A
    /dev/mchanger0: STK L700 V-1012 Target-3 LUN-0
    /dev/mchanger1: STK L700 V-1012 Target-3 LUN-1

    Target-4 WWNN-50-01-04-F0-00-7E-B3-45 WWPN-50-01-04-F0-00-7E-B3-46
    /dev/mchanger2: STK VLIBRARY V-1012 Target-4 LUN-1
    /dev/mchanger3: STK VLIBRARY V-1012 Target-4 LUN-2

LSILogic 53c1030  LSILogic,scsi@1
    /dev/mchanger4: STK SL500 Target-0 LUN-0

Emulex LP10000  fibre-channel@2
    HBA:  WWNN 20:00:00:00:c9:46:57:b3  WWPN 10:00:00:00:c9:46:57:b3

    Target-0 WWNN-50:01:04:F0:00:41:18:46 WWPN-50:01:04:F0:00:41:19:32
    /dev/mchanger5: STK L180 V-0310 Target-0 LUN-0

# su - acsss
$ whoami
acsss

$ showDevs.sh -sq
    /dev/mchanger0: STK V700 S/N-00043000022 216-cells 10-drives
    /dev/mchanger1: VL700 S/N-00043000040 10-drives in use by ACSLS.
    /dev/mchanger2: STK VLIBRARY S/N-00043000037 1000-cells 50-drives
    /dev/mchanger3: STK VLIBRARY S/N-00043000045 10000-cells 461-drives
    /dev/mchanger4: SL500 S/N-52200000032 6-drives in use by ACSLS.
    /dev/mchanger5: L180 S/N-MPC01000124 4-drives in use by ACSLS.

$ showDevs.sh -s
QLogic QLA2300  QLGC,qla@1
    /dev/mchanger0: STK V700 S/N-00043000022 216-cells 10-drives
    /dev/mchanger1: VL700 S/N-00043000040 10-drives in use by ACSLS.
    /dev/mchanger2: STK VLIBRARY S/N-00043000037 1000-cells 50-drives
    /dev/mchanger3: STK VLIBRARY S/N-00043000045 10000-cells 461-drives

LSILogic 53c1030  LSILogic,scsi@1
    /dev/mchanger4: SL500 S/N-52200000032 6-drives in use by ACSLS.

Emulex LP10000  fibre-channel@2
    /dev/mchanger5: L180 S/N-MPC01000124 4-drives in use by ACSLS.

```

## ■ stats\_report

The `stats_report` utility generates library volume statistics reports. To run this utility, you must be logged in as `acsss`.

### Format

```
stats_report
```

## Usage

- The `stats_report` uses the current `acsss_stats.log` to prepare two reports of volume statistics. You must enable library volume statistics gathering by setting the variable `LIB_VOL_STATS` to on. This can be done through the `acsss_config` process or through the command line command `dv_config -p LIB_VOL_STATS`. ACSLS then automatically rolls and maintains 9 `acsss_stats.log` files when the log reaches the default size of 500 KB.
- The size of the log files and the number of files to retain is controlled through the variables `LIB_STATS_FILE_NUM` and `VOL_STATS_FILE_SIZE`. These variables are set using the same method as `LIB_VOL_STATS` discussed above.
- The two types of reports are:
  - `drive_centric.txt`

This report contains an ordered list of drives. Each drive record contains all cartridges mounted to the drive, the requestor, the time of the request and the duration of the mount.

- `time_centric.txt`

This report contains the usage of drive resources listed on an hourly time scale. Each record in a time period includes the requestor, the specific drive, the number of mounts during that period for that drive, and the duration of drive usage during the hour. Note: if the drive usage exceeds 60 minutes for a time period it is an indication that the mount spanned two time periods and the mount will not be listed in the second time period. The first report created by `stats_report` is a drive view.

## ■ volrpt

The `volrpt` utility creates a volume report.

### Format

```
volrpt [-s vol|loc|use] [-d] [-f filename] [-z] [-a|-l|-v  
identifier_list] [-i]
```

### Options

-s

specifies the sort order. If you do not specify this option, the default is to sort by volume ID. If you specify this option, you must specify one of the following values:

- `vol`  
sort by volume ID.
- `loc`  
sort by volume home location.
- `use`  
sort by volume use (number of mounts).
- `-d`  
specifies that the output contains no page breaks or header information. The output can be used as input to other programs such as `pr`.
- `-f filename`  
*filename* specifies a custom `volrpt` template.
- `-Z`  
zero fills identifier fields.
- `-a`  
restricts the report to the specified ACS. You can specify multiple ACSs (use blanks to separate the *acs\_ids*).
- `-l`  
restricts the report to the specified LSM. You can specify multiple LSMs (use blanks to separate the *lsm\_ids*).
- `-v`  
restricts the report to the specified volumes (or volume ranges). You can specify a single *vol\_id*, a list of *vol\_ids* separated by blanks, or a volume range indicated by *vol\_id-vol\_id*.
- `-identifier_list`  
described by the `-v`, `-a`, and `-l` options. This is a list of ACSs, LSMs, and volumes (or volume ranges).
- `-i`  
reports all volumes, including absent and ejected cartridges.  
  
If this option is not specified, absent and ejected cartridges are not reported.

## Usage

Use the `volrpt` utility to create a report of library cartridges, including their physical location, history, attributes, and use. You can also use `volrpt` to verify the database after you restore it. You can use the `-a`, `-l`, or `-v` options to specify the ACSs, LSMs, or cartridges for the report. If you do not specify any of these options, `volrpt` reports on only ACS 0.

Figure 44 shows an example of a standard volume report, which contains fields for volume id, location, label type, media type, and history of usage.

**Figure 44. Standard Volume Report**

```

                                VOLUME REPORT UTILITY

2002-06-30 14:01:21

TOTAL VOLUMES: 400 SEQUENCE: sort by volume identifier

Volume Home   Label      Volume      Times      |---Entered---||--Last Used--|
Label  Location Attr   Type/Media  Mounted    Date       Time       Date       Time
-----
CLN000 0,0,1,0,3 Ext   C/STK1U     1           08/22/01  09:30      10/04/01  14:26
RB0000 0,1,2,1,10Ext. D/STK1R     3           10/01/01  08:16      10/01/01  08:18
RB1400 0,0,10,1,3Ext. S/STK1R    243         10/01/01  09:30      10/06/01  11:04
RB1401 0,0,10,3,5Virt. D/STK1R    12          10/01/01  03:29      10/05/01  23:11
"      "      "      "      "      "      "      "
"      "      "      "      "      "      "      "
"      "      "      "      "      "      "      "
TB1440 0,1,3,1,9 Ext.   D/STK2P     43          08/12/01  09:11      09/28/01  17:52
"      "      "      "      "      "      "      "
"      "      "      "      "      "      "      "
"      "      "      "      "      "      "      "

```

In the Volume Type/Media column, C denotes cleaning cartridges, D denotes data cartridges, and S denotes scratch cartridges.

Use the `-f filename` option to create a customized report; see [“Creating a Custom Volume Report” on page 233](#), for more information.

`$ACS_HOME/data/external/volrpt/owner_id.volrpt` is a sample input file that you can run or use as a template to create customized volume reports. You can also save your customized volume reports in the `$ACS_HOME/data/external/volrpt` directory.

You can redirect the volume report to a file with standard UNIX redirection:

```
volrpt > file
```

## Examples

By default, volrpt reports only the first ACS in the list. To report the cartridges in both ACS 0 and ACS 1, enter:

```
volrpt -a 0 1
```

To report the cartridges in LSMs 0,1 and 2,1 sorted by home cell location, enter:

```
volrpt -s loc -l 0,1 2,1
```

## Notes

- `volrpt` displays the specified volume report if it completes successfully. `volrpt` prints a message to `stderr` and exits if you specify the `-f` option and `volrpt` cannot find the specified file or you specify more than one input file. For field errors within the input file, `volrpt` prints a message to `stderr` and ignores the line in error but does not exit.
- If cartridges are not found in the volume ID list or range or library component you specified, `volrpt` returns a `no volumes found` message.
- When a parameter is not specified, it uses the default of ACS 0.
- If a library component(s) is specified through the `-a`, `-l`, or `-v` option, but no volumes are found, messages such as the following are displayed:

- **-a option (ACS)**

Messages:

when a single `acs_id` is provided and no volumes are present, the following error displays: No Volumes found for ACS:  
(`<acsid>`)

Example:

```
$ volrpt -a 2
No Volumes found for ACS: (2)
```

when multiple `acs_ids` are provided and none of them have any volumes, the following error displays: No Volumes found for ACS: (`<acsid1>`)(`<acsid2>`)

Example:

```
$ volrpt -a 0 1
No Volumes found for LSM: (0) (1)
```

- **-l option (LSM)**

Messages:

when a single `lsm_id` is provided and no volumes are present, the following error displays: No Volumes found for LSM:  
(`<lsmid>`)

Example:

```
$ volrpt -l 1,1
No Volumes found for LSM: (1,1)
```

when multiple lsm\_ids are provided and none of them have any volumes, the following error displays: No Volumes found for LSM: (<lsmid1>)(<lsmid2>)

Example:

```
$ volrpt -l 1,1 1,2
No Volumes found for LSM: (1,1) (1,2)
```

- **-v option (VOLUME)**

Messages:

when a single valid is provided and no volumes are present, the following error displays: Volume(s) not: (<valid>)

Example:

```
$ volrpt -v BBB112
No Volumes found: (BBB112)
```

when multiple valids are provided and none of them have any volumes, the following error displays: Volume(s) not found: (<valid1>)(<valid2>)

Example:

```
$ volrpt -v BBB112 BBB114
No Volumes found: (BBB112) (BBB114)
```

The **-v** option can also be used for volume range, and produces similar messages when no volumes are present.

when a single volume range is provided and no volumes are present, the following error displays: Volume(s) not: (<volrange>)

Example:

```
$ volrpt -v BBB112-BBB116
No Volumes found: (BBB112-BBB116)
```

when multiple volume range is provided and no volumes are present, the following error displays: Volume(s) not: (<volrange1>) (<volrange2>)

Example:

```
$ volrpt -v BBB112-BBB116 BBB220-BBB224
No Volumes found: (BBB112-BBB116) (BBB220-BBB224)
```

- When an ACS or LSM has not been configured

When `volrpt` is used with an `acs_id` or `lsm_id` that does not exist, it displays a message according to the identifier.

- **-a** (ACS)

```
ACS identifier (<acsid>) not configured
```

- **-l** (LSM)

```
LSM identifier (<lsmid>) not configured
```

## See Also

- [“Creating a Custom Volume Report” on page 233](#)

## ■ watch\_vols

This utility applies a pre-defined policy to newly entered or newly discovered volumes in an ACSLS library. That policy is defined in the file:

```
$ACS_HOME/data/external/vol_attr.dat
```

This file contains a list of user-defined volume IDs or volume ranges and a user-specified policy with each volume entry. For each volume or volume range listed in that file, you can define volume ownership, pool association and/or preferred LSM location. Specific instructions for defining policies are explained in detail in `vol_attr.dat`.

The `watch_vols` utility uses the `acsss_stats.log` to discover the existence of newly entered volumes or volumes discovered during an audit. To enable this capability, you must enable volume statistics with `acsss_config`. With volume statistics enabled, `watch_vols` monitors the tail of the `acsss_stats.log`, looking for matching volumes with the entries defined in `vol_attr.dat`. Wherever a match is found, the defined policy for that volume is automatically applied.

## Format

```
watch_vols [start|stop]
```

## Usage

You can check the running status of the utility by invoking `watch_vols` with no parameter. If you are unsure of the status of `watch_vols` (running or stopped), the command `watch_vols` with no argument will display the current status.

There are two options for `watch_vols`, *start* and *stop*.

- `watch_vols start`

When the `start` parameter is invoked, `watch_vols` reviews the policy defined in `vol_attr.dat`. If there are errors in format or syntax, `watch_vols` displays the error and prompts you to make the necessary correction to `vol_attr.dat`. Once the defined policy is accepted by `watch_vols`, the utility invokes a daemon to run in the background. The daemon continues to run as long as ACSLS is running. It starts automatically whenever ACSLS is restarted.

The policy table in `vol_attr.dat` can be updated at any time. You need not stop `watch_vols` in order to update the policy. Just run `watch_vols start` to commit the updates to the running program.

- `watch_vols stop`

A log of all `watch_vols` activities is maintained in the log file

**`$ACS_HOME/log/watch_vols_event.log`**

Each change of volume ownership, `pool_id`, or LSM home location is logged in this file.

## Examples

- You want to move specific volumes to a target LSM during an initial audit
  1. Audit the target LSM with `watch_vols` **disabled**.
  2. Once the target LSM has been audited, start `watch_vols`.
  3. Perform a general audit.Audit then migrates the specified volumes to the destination LSM.



---

In this chapter, you will learn how to use the ACSLS commands, including general command syntax and reference information.

[“audit” on page 294](#)

Creates or updates the database inventory of the volumes in a library component.

[“cancel” on page 300](#)

Cancels a current or pending request.

[“clear lock” on page 304](#)

Removes all active and pending locks on transports or cartridges.

[“define pool” on page 306](#)

Creates or modifies scratch pools.

[“delete pool” on page 308](#)

Deletes empty scratch pools.

[“dismount” on page 310](#)

Dismounts a cartridge.

[“eject” on page 314](#)

Ejects one or more cartridges from the ACS.

[“enter” on page 318](#)

Sets a CAP to enter mode.

[“idle” on page 322](#)

Stops ACSLS from processing new requests.

[“lock” on page 324](#)

Locks (dedicates) a cartridge or transport to a user.

[“logoff” on page 326](#)

Exits from cmd\_proc.

[“move” on page 335](#)

moves a specified cartridge to an available storage cell in a specified LSM.

[“mount” on page 327](#)

Mounts a data or scratch cartridge.

[“query commands” on page 337](#)

Displays the status of a library component.

[“set commands” on page 364](#)

Sets various attributes of different library components.

[“show” on page 375](#)

Displays your lock ID or user ID.

[“start” on page 376](#)

Starts ACSLS request processing.

[“switch lmu” on page 377](#)

Manually switches ACS management from the ACS's master LMU to the standby LMU.

[“unlock” on page 380](#)

Removes active locks on cartridges or drives.

[“vary” on page 382](#)

Changes the state of an ACS, LSM, CAP, transport, or port.

[“venter” on page 388](#)

Enters one or more cartridges with missing or unreadable labels into the ACS.

**Note:** The L5500, SL500, and SL8500 libraries do not support unlabeled cartridges (venters).

## ■ General Command Syntax

This section describes the general syntax of the ACSLS commands. The following sections fully describe each command, including its syntax.

ACSLS commands use the following general syntax:

*command type identifier state options*

Where:

*type identifier*

is the ACS component and its identifier; see [“Component Types and Identifiers” on page 291](#) for more information.

*state*

is a device state for the `vary` command only.

*options*

are command options; see the description of the command you want to run.

Also note the following syntax rules:

- You must enter commands in the order shown above (command name, followed by the component and its identifier, followed by the state and any options).

- This chapter shows commands in lowercase, but you can enter commands in any combination of lowercase and uppercase letters.
- Underlines show minimum abbreviations for commands and keywords. For example, `query server` and `q ser` are both valid forms of the `query server` command.
- Ellipses (...) indicate that you can repeat an identifier.
- Brackets [ ] enclose optional options.

## ■ Component Types and Identifiers

[Table 23](#) describes the valid ranges of values for each of the ACS component identifiers. You can specify a maximum of 42 identifiers for each type. See specific command descriptions for the valid component types for each command.

**Note:** The identifiers specified in [Table 23](#) represent valid ranges of values supported by the software. Your LSM type and library configuration determine what identifier values are valid for your particular site.

**Table 23. ACSLS Component Types and Identifiers**

Component	Type	Identifier	Valid Values
entire library	<u>server</u>	none	none
ACS	<u>acs</u>	<i>acs_id</i>	acs(0-126), lsm(0-23)
LSM	<u>lsm</u>	<i>lsm_id</i>	acs(0-126), lsm(0-23)
LSM panel	<u>panel</u>	<i>panel_id</i>	acs(0-126), lsm(0-23), panel(0-19),
LSM subpanel *	<u>subpanel</u>	<i>subpanel_id</i>	acs(0-126), lsm(0-23), panel(0-19), startrow(0-41), startcolumn(0-23), endrow(0-41), endcolumn(0-23)

**Table 23. ACSLS Component Types and Identifiers**

Component	Type	Identifier	Valid Values
LSM storage cell	<u>subpanel</u>	<i>cell_id</i>	acs(0-126), lsm(0-23), panel(0-19), row(0-41), column(0-23)

\* The ending row (and column) must be greater than or equal to the beginning row (and column). Only cells within the beginning and ending matrix are audited: the matrix starts with the beginning row and beginning column and extends to the ending row and ending column.

CAP	<u>cap</u>	<i>cap_id</i>	acs(0-126), lsm(0-23), cap(0-2)  An asterisk (*) in the <i>cap_id</i> does the following:  acs,lsm,* - causes ACSLS to select the highest priority available CAP in the LSM.  acs,* - causes ACSLS to select the highest priority available CAP in the ACS.  * - for an enter request causes ACSLS to select the CAP in the LSM with the most free cells.  * - for an eject request causes ACSLS to select the highest priority CAP in each ACS with a cartridge designated for ejection.
-----	------------	---------------	---

**Table 23. ACSLS Component Types and Identifiers**

<b>Component</b>	<b>Type</b>	<b>Identifier</b>	<b>Valid Values</b>
transport	<u>drive</u>	<i>drive_id</i>	acs(0-126), lsm(0-23), panel(0-19), drive(0-19)
transport type	<u>drive</u>	<i>drive_type</i>	10-character transport type identifier; can be any combination of numbers (0-9) or letters (A-Z). Spaces are not allowed.
port	<u>port</u>	<i>port_id</i>	acs(0-126), port(0-15)
volume serial number of a data or scratch cartridge or a cleaning cartridge	<u>volume</u> , <u>scratch</u> , <u>clean</u>	<i>vol_id</i>	Six-character identifier consisting of any combination of numbers (0-9), letters (A-Z, a-z or mixed case), dollar sign (\$), pound sign (£), and spaces ( ). Use single or double quotes to enclose volsers with leading or trailing spaces. <i>Do not</i> specify volsers with embedded spaces.
range of volume serial numbers	<u>volume</u>	<i>volrange</i>	Specifies an ascending range of volumes separated by a dash (-). Specify only the right-most numeric portions of the volsers as the range. All preceding characters must be identical.

**Table 23. ACSLS Component Types and Identifiers**

Component	Type	Identifier	Valid Values
volume media type	<u>m</u> edia	<i>media_type</i>	10-character media type identifier; can be any combination of numbers (0-9) or letters (A-Z). Spaces are not allowed.
volume owner	<u>o</u> wner	<i>owner_id</i>	
scratch pool	<u>p</u> ool	<i>pool_id</i>	Decimal number (0-65534). Specifying an asterisk (*) for the <i>pool_id</i> reassigns the volume to its current <i>pool_id</i> .
ACSLs request	<u>r</u> equest	<i>request_id</i>	Unique decimal number (0-65535) assigned by ACSLS.
transport or volume lock	<u>l</u> ock	<i>lock_id</i>	Decimal number (0-32767)
ACSLs request	<u>r</u> equest	<i>request_id</i>	unique numeric (0-65535) request identifier assigned by ACSLS.

## ■ Common Command Error Messages

A common error message appears if ACSLS rejects a command because of a syntax error, invalid identifier, type, or option, process failure, database error, and so forth. For more information on common error messages, see *ACSLs Messages*.

If you enter a command that you cannot use, the following message appears:

```
Command access denied.
```

If you specify a volume that you cannot access, the following message appears:

```
Volume access denied.
```

## ■ audit

The audit command updates the ACSLS database to match the actual inventory of library cartridges.

**Note:** After a SL8500 has been expanded, and you are auditing it for the first time, perform the following steps to avoid losing cartridge information:

1. Audit the panels in the section attached to the CAP(s) that were moved to accommodate the new expansion section(s).
2. Audit the remaining panels.

### **Format**

*audit cap\_id type identifier...*

*cap\_id*

specifies the CAP that ACSLS uses to eject any errant volumes. You can specify a particular cap or enter an asterisk (\*), which causes ACSLS to select the highest priority CAP.

You can specify multiple ACSs only if CAP priority has been set in each ACS. You must specify an asterisk for the *cap\_id* to allow automatic selection of a CAP in each ACS.

A single LSM audit waits until ACSLS updates the database, then reserves the CAP if required to eject errant volumes.

*type identifier*

specifies a library component. [Table 24](#) lists the components you can audit.

**Table 24. Valid Components for Audit**

<b>Library Component</b>	<b>type</b>	<b>identifier</b>
entire library	<i>server</i>	<i>none</i>
ACS	<i>acs</i>	<i>acs_id</i>
LSM	<i>lsm</i>	<i>lsm_id</i>
LSM panel	<i>panel</i>	<i>panel_id</i>
LSM subpanel	<i>subpanel</i>	<i>subpanel_id</i>

You can specify multiple ACSs, LSMs, panels, or subpanels in a single audit request. You cannot specify overlapping subpanels.

You can specify multiple ACSs only if CAP priority has been set in each ACS. You must specify an asterisk for the *cap\_id* to allow automatic selection of a CAP in each ACS.

When you audit the server or an ACS, the parent audit process creates a separate audit process for each LSM. These LSM audits run in parallel and process panel-by-panel in ascending *panel\_id* sequence. To audit all LSMs in all libraries or an ACS, just audit the server or ACS. There is no need to run multiple, separate LSM audits.

However, within an LSM, regardless of the order in which you specify multiple components the audit processes these components in ascending order by *acs\_id*, *lsm\_id* *panel\_id*, *subpanel\_id*. For example, the audit will process panel 0,0,9 before subpanel 0,0,10,1,7 even if you specify the subpanel first.

### **Usage**

An audit updates the ACSLS database to match the actual inventory of library cartridges. You run an audit to:

- Create volume information in the database if you do not enter cartridges through the CAP. For example, run an audit if you add an LSM to your library, open the LSM door, and manually add cartridges to the LSM.
- Resolve discrepancies between the library and the database. For example, run an audit if you open an LSM door and manually remove cartridges instead of ejecting them through the CAP. The audit deletes information for the removed cartridges from the database.
- View the contents of cells specified by the audit (you must have a display monitor connected to the LSM robot's vision system).

**Note:** Audit recognizes particular models of cleaning cartridges and will record them in the database as cleaning cartridges. However, the audit command does not set the *max\_uses* for new cleaning cartridges it encounters. You will have to set the *max\_uses* for these using the set clean command on pages 160 and 300.

When audit cannot find a cartridge in any recorded location (cell and possibly drive), audit either marks the cartridge as absent (if absent volume retention is enabled) or deletes the volume. When audit deletes the volume, it removes all information about the volume from the database, including volume information and customer-supplied information about access control, scratch status, lock IDs, and pool IDs. If audit later finds the cartridge in another location, it re-adds the volume information, but the customer-supplied information is lost.

An audit ejects errant volumes and deletes their information from the database. An errant volume has:

- An external label that duplicates one already scanned.
- A missing or unreadable external label and no virtual label.
- An invalid media type.

Audits only apply to LSM storage cells, not to transports or CAPs. ACSLS displays `cmd_proc` messages during the audit and records any database changes from the audit in the event log. If volume statistics logging is enabled, ACSLS records additional volume found and volume not found messages in the `acsss_stats.log`. See [“Creating a Logging Volume Statistics Report” on page 236](#) for more information.

**Hint:** Use the following guidelines for running an audit:

- The ACS or LSM being audited must be either online or in diagnostic state. Normal library processing (including mounts and dismounts) can occur during an audit, although library processing slows down the audit.

The difference between online and diagnostic states is: an audit in diagnostic state can only be performed through the `cmd_proc`. Client requests are rejected while the LSM is in the diagnostic state. This state is faster because the audit now has exclusive run of the library without competing mount/dismount requests that could otherwise be submitted from the client.

- After you start an audit on an entire LSM, you cannot start another audit on the same LSM. You must cancel and rerun the audit.
- You can cancel any audit request, but ACSLS will always finish auditing the current panel or subpanel. When you cancel an audit, some or all of the cartridges marked for ejection may not be ejected. When you cancel an audit, cartridges already ejected are not re-entered.

**CAUTION:** If you cancel an audit or if there is a library or ACSLS hardware or software failure during the audit, you must rerun the same audit. Cartridges marked for ejection but not actually ejected during the first audit are no longer in the database and are not under ACSLS control.

### **Examples**

To audit the entire library and specify the highest priority CAP in each ACS for ejections:

```
audit * server
```

To audit LSM 0,1 and specify CAP 0,1,1 for ejections:

```
audit 0,1,1 lsm 0,x
```

To audit panel 10 of LSM 0,1 and specify the highest priority CAP in LSM 0,1 for ejections:

```
audit 0,1,* panel 0,1,10
```

### **Notes**

A cell cannot be audited if it is reserved by another process. If a cell is reserved, ACSLS rechecks the database until the cell becomes available up to a maximum of 60 retries. If the cell is still unavailable, the audit skips the cell and logs a message to the event log.

**See Also**

<b>For information about...</b>	<b>See...</b>
Guidelines for auditing the library	<a href="#">“Auditing the Library” on page 163</a>
Cancelling a command	<a href="#">“cancel” on page 300</a>
Displaying CAP status	<a href="#">“query cap” on page 340</a>
Displaying ACSLS and library status	<a href="#">“query server” on page 361</a>
Displaying ACS status	<a href="#">“query acs” on page 338</a>
Displaying LSM status	<a href="#">“query lsm” on page 349</a>
Setting CAP selection priority	<a href="#">“set cap priority” on page 367</a>
Setting CAP mode (manual or automatic)	<a href="#">“set cap mode” on page 365</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>

**Command Area Messages****Success Messages**

The following message appears when the audit completes successfully.

```
Audit: Audit completed, Success.
```

In addition, one of the following messages appears to confirm which component was audited:

```
Audit: Audit of storage server, valid
Audit: Audit of ACS, acs_id, status valid
Audit: Audit of LSM, lsm_id, panel_id, valid
Audit: Audit of panel, panel_id, valid
Audit: Audit of subpanel, subpanel_id, valid
```

**Hint:** If you audit an invalid cell location (that is, just above or below a transport or on an extra column), ACSLS returns a success message to allow audits to work with PTPs.

## Intermediate Messages

Intermediate messages consist of a two-line display with the following first line:

```
Audit: Intermediate response: Audit activity.
```

One of the following messages appears on the second line.

```
Audit: Volume ejected, unreadable label.
```

**Explanation:** ACSLS ejected a cartridge that had:

- No external label
- No virtual label, or
- An unreadable label

```
Audit: Volume vol_id ejected, duplicate label.
```

**Explanation:** ACSLS ejected with a duplicate external label within the range of cells being audited.

**Variable:** *vol\_id* is the volume with the duplicate label.

```
Audit: Volume vol_id found.
```

**Explanation:** The audit found a volume in the ACS that is not in the ACSLS database. The audit added the volume to the database.

**Variable:** *vol\_id* is the volume added to the database.

```
Audit: Volume vol_id not found.
```

**Explanation:** A volume listed in the ACSLS database is not in the ACS. The volume is deleted from the database.

**Variable:** *vol\_id* is the volume deleted from the database.

```
Audit: Volume ejected, invalid media type
```

**Explanation:** ACSLS ejected a volume with an invalid media type.

## Error Messages

```
Audit in progress.
```

**Explanation:** ACSLS did not start the audit because another audit for the same LSM is in progress.

```
CAP cap_id in use.
```

**Explanation:** The CAP specified for the audit is in use.

**Variable:** *cap\_id* is the CAP in use.

Multiple ACS audit.

**Explanation:** The audit failed because the audit command specified multiple ACSs without specifying an \* for the *cap\_id*.

Not in same ACS.

**Explanation:** The audit failed because the *cap\_id* and the *identifier* specified are not in the same ACS.

**Variable:**

- *cap\_id* is the CAP specified for the audit.
- *identifier* is the library component specified for the audit.

### **Display Area Messages**

*cap\_id* Remove cartridges from CAP.

**Explanation:** The audit filled the CAP with ejected cartridges. Empty the CAP, then close it to continue the audit.

**Variable:** *cap\_id* is the CAP that contains the ejected cartridges.

CAP *cap\_id*: Place magazines in CAP.

**Explanation:** The CAP requires magazines for the audit. Open the CAP, place magazines inside, then close the CAP.

**Variable:** *cap\_id* is the CAP that requires magazines.

CAP *cap\_id*: No CAP available, waiting...

**Explanation:** No CAP is available to eject cartridges.

**Variable:** The *cap\_id* appears as it was specified in the audit command:

- *acs, lsm, cap* if the `audit` command explicitly specified the CAP.
- *acs, lsm, \** if the `audit` command specified the CAP as *acs, lsm, \**.
- *acs, \*, \** if the `audit` command specified the CAP as *acs, \** or *\**.

## ■ **cancel**

The `cancel` command cancels a current or pending request.

### **Format**

`cancel request_id`

### **Options**

*request\_id*

specifies the identifier of the request to cancel.

Use the `cancel` command to cancel current or pending request issued by an `audit`, `define pool`, `delete pool`, `eject`, `enter`, `lock`, `query`, `set`, or `venter` command or client application. Use the query `request` command to display the ID of the request you want to cancel.

You can cancel an audit of a server, ACS, or LSM. Because ACSLS internally translates server, ACS, or LSM audits into a series of panel audits, ACSLS completes the audit of the current panel before cancelling the remainder of the audit. You cannot cancel an audit of a panel or subpanel. When you cancel an audit, cartridges already ejected are not re-entered.

**CAUTION:** If you cancel an audit or if there is a library or ACSLS hardware or software failure during the audit, you must rerun the same audit. Cartridges marked for ejection but not actually ejected during the first audit are no longer in the database and are not under ACSLS control.

**Hint:** You must enter a `cancel` command from a different `cmd_proc` than the `cmd_proc` that issued the request you want to cancel.

The `cancel` command immediately cancels any pending requests and handles current requests as follows:

#### `audit`

Because ACSLS internally translates server, ACS, or LSM audits into a series of panel audits, ACSLS completes the audit of the current panel before cancelling the remainder of the audit.

**CAUTION:** If you cancel an audit or if there is a library or ACSLS hardware or software failure during the audit, you must rerun the same audit. Cartridges marked for ejection but not actually ejected during the first audit are no longer in the database and are not under ACSLS control.

#### `define pool`

ACSLs stops defining scratch pools but does not delete any scratch pools already defined.

#### `delete pool`

ACSLs stops deleting scratch pools but does not redefine any scratch pools already deleted.

#### `eject`

ACSLs stops the ejection and the `cmd_proc` displays a message to remove any cartridges already ejected, which are not reentered.

The `eject` does not terminate until the cartridges are removed, the CAP is closed, and ACSLS verifies that the CAP is empty.

#### `enter`

ACSL S stops the enter and, if any cartridges remain in the CAP, `cmd_proc` displays a message to remove these cartridges. Any cartridges already entered into the LSM are not ejected.

The enter does not terminate until the cartridges are removed, the CAP is closed, and ACSLS verifies that the CAP is empty.

**Note: Automatic enter.** You cannot cancel an automatic enter operation that is in progress using the cancel command. To terminate an automatic enter in progress:

- If the CAP door is open, remove all the cartridges and close the door.
- If the CAP door is closed and the cartridges are being moved into the library, you must allow the remaining cartridges to be entered into the library. The enter then terminates.

lock

Resource locking by the specified request is stopped. If the request has not yet acquired all specified resources, none of the resources are locked.

query

ACSL S cancels the query.

set

For set cap, ACSLS stops setting CAP attributes, but does not change any attributes already set.

For set scratch or set clean requests, ACSLS stops setting scratch cartridge or cleaning cartridge attributes, but does not change any attributes already set.

venter

ACSL S stops the enter and, if any cartridges remain in the CAP, `cmd_proc` displays a message to remove these cartridges. Any cartridges already entered into the LSM are not ejected.

**Examples**

To display request IDs for all current and pending requests:

```
query request all
```

Example output of `query request all`:

Identifier	Command	Status
13	enter	Current
15	query	Pending

To cancel request 13 (current enter request) in the example above:

```
cancel 13
```

**Notes**

None.

**See Also**

For information about...	See...
Updating the ACSLS database to match the actual inventory of library cartridges	<a href="#">“audit” on page 294</a>
Creating or modifying scratch pools	<a href="#">“define pool” on page 306</a>
Deleting empty scratch pools	<a href="#">“delete pool” on page 308</a>
Ejecting cartridges from the library	<a href="#">“eject” on page 314</a>
Making a CAP (manual mode) ready to enter labelled cartridges into the library	<a href="#">“enter” on page 318</a>
Locking (dedicating) drives and cartridges to your current lock ID	<a href="#">“lock” on page 324</a>
Displaying the status of a library component	<a href="#">“query commands” on page 337</a>
Setting various attributes of different library components	<a href="#">“set commands” on page 364</a>
Making a CAP ready to enter unlabeled cartridges into the library	<a href="#">“venter” on page 388</a>

**Command Area Messages****Success Messages**

Request *request\_id* cancelled.

**Explanation:** ACSLS cancelled the requested command.

**Variable:** *request\_id* is the request identifier of the cancelled command.

**Intermediate Messages**

None.

**Error Messages**

Request *request\_id* can not be cancelled: *status*.

**Explanation:** ACSLS cannot cancel the specified command.

**Variable:**

- *request\_id* is the request identifier of the command that ACSLS cannot cancel.

- *status* is one of the following:

Request identifier *request\_id* invalid.

The cancel command specified an invalid request identifier.

Request identifier *request\_id* not found.

The cancel command specified an request identifier for a request that is not current or pending.

### **Display Area Messages**

None.

## ■ **clear lock**

The `clear lock` command removes all active and pending locks on a specified drive or cartridge.

### **Format**

`clear lock type identifier`

### **Options**

*type identifier*

specifies a library component. [Table 25](#) lists the components whose resource locks you can clear.

**Table 25. Valid Components for Clear Lock**

<b>Library Component</b>	<b>type</b>	<b>identifier</b>
transport	drive	drive_id
volume	volume	vol_id

### **Usage**

Use the `clear lock` command to remove all active and pending locks on a specified drive or cartridge. Your current lock ID must either be 0 or it must match the lock ID of the driver or cartridge.

The `unlock` command removes only active locks on drives or cartridges. You can, however, use the `unlock` command to remove active locks on all drives or all cartridges.

**Note:** The `clear lock` command always resets your lock ID to 0.

### **Examples**

To clear all locks for transport 1,1,5,2:

```
clear lock drive 1,1,5,2
```

To clear all locks for volume NN0108:

```
clear lock volume NN0108
```

### Notes

You cannot cancel a clear lock command.

### See Also

For information about...	See...
Locking drives and cartridges	<a href="#">“lock” on page 324</a>
Displaying the lock status of a drive or cartridge	<a href="#">“query lock” on page 347</a>
Setting your lock ID	<a href="#">“set lock” on page 370</a>
Displaying your lock or user ID	<a href="#">“show” on page 375</a>
Removing active locks for drives or cartridges	<a href="#">“unlock” on page 380</a>

### Command Area Messages

#### Success Messages

The following message appears when a clear request succeeds:

```
Clear: Completed, Success.
```

In addition, for each identifier in the request, one of the following messages appears depending on the library component (*type*):

```
Clear: Drive drive_id all locks cleared.
Clear: Volume vol_id all locks cleared.
```

#### Intermediate Messages

None.

#### Error Messages

```
Clear: Clear lock of drive drive_id failed,
Drive identifier drive_id available.
```

**Explanation:** ACSLS cannot clear locks because the specified transport is not locked.

**Variable:** *drive\_id* is the identifier of the specified transport.

```
Clear: Clear lock of volume vol_id failed,
Volume identifier vol_id available.
```

**Explanation:** ACSLS cannot clear locks because the specified volume is not locked.

**Variable:** `vol_id` is the identifier of the specified volume.

### ***Display Area Messages***

None.

## ■ **define pool**

The `define pool` command creates or modifies scratch pools.

### ***Format***

```
define pool low_water_mark high_water_mark
pool_id...[overflow]
```

### ***Options***

*low\_water\_mark*

is the low volume warning threshold. If the scratch cartridge count falls below this threshold, ACSLS logs a warning message in the event log. Valid values are 0 to  $2^{31}-1$ . The default is 0.

*high\_water\_mark*

is the high volume warning threshold. If the scratch cartridge count reaches or exceeds this threshold, ACSLS logs a warning message in the event log. This value must be greater than the value for *low\_water\_mark*.

*pool\_id*

specifies the pool identifier. Pool 0 is the common scratch pool, which always exists. You can modify the common scratch pool attributes.

*overflow*

specifies that if this pool cannot satisfy `mount scratch` requests, ACSLS will select cartridges from the common pool (Pool 0).

### ***Usage***

Use the `define pool` command to create or modify scratch pools.

### ***Examples***

To define a low threshold of 0, a high threshold of 600, and overflow for new pool 1:

```
define pool 0 600 1 overflow
```

To define a low threshold of 0, a high threshold of 600, and no overflow for existing pool 5:

```
define pool 0 600 5
```

**Notes**

Scratch pools are not owned by one client application or user ID. You can, however, use volume access control to restrict access for specific scratch cartridges.

**See Also**

For information about...	See...
Cancelling a command	<a href="#">“cancel” on page 300</a>
Deleting empty scratch pools	<a href="#">“delete pool” on page 308</a>
Guidelines and procedures for managing scratch cartridges	<a href="#">“Managing Scratch Cartridges” on page 195</a>
Mounting a scratch cartridge on a transport	<a href="#">“mount *” on page 330</a>
Displaying scratch pool attributes	<a href="#">“query pool” on page 355</a>
Setting or clearing cartridge scratch attributes	<a href="#">“set scratch” on page 372</a>

**Command Area Messages****Success Messages**

The following message appears when a define pool command succeeds:

```
Define: Define completed, Success.
```

In addition, the following message appears for each created or modified pool:

```
Define: Pool pool_id created.
```

**Intermediate Messages**

None.

**Error Messages**

None.

**Display Area Messages**

```
Pool pool_id: low water mark warning.
```

**Explanation:** The number of volumes in the specified scratch pool is less than or equal to the low volume threshold.

**Variable:** *low\_water\_mark* is the low volume threshold of the specified scratch pool.

```
Pool pool_id: high water mark warning.
```

**Explanation:** The number of cartridges in the specified scratch pool is greater than or equal to the high volume threshold.

**Variable:** *high\_water\_mark* is the high volume threshold of the specified scratch pool.

## ■ delete pool

The `delete pool` command deletes empty scratch pools.

### **Format**

```
delete pool pool_id...|all
```

### **Options**

*pool\_id*  
specifies the pool ID.

Pool 0 is the common pool, which you cannot delete.

*all*  
specifies all empty scratch pools.

### **Usage**

Use the `delete pool` command to delete empty scratch pools. If a pool contains scratch cartridges, you must reassign these cartridges to another pool before deleting the first pool. If a scratch cartridge is mounted, it becomes a data cartridge but remains in its scratch pool. Use the `set scratch off` command to reassign the data cartridge to the common pool.

### **Examples**

To delete all empty scratch pools:

```
delete pool all
```

**Hint:** Only the empty pools will be deleted; pools with cartridges assigned will not be affected.

To delete scratch pool 1:

## 1. Query scratch pool 1:

```
query scratch 1
```

```
1998-06-30 09:35:30 Scratch Status
Scratch Pool Identifier Home Location Status Type
1 34813 0,0,1,8,8 home 3480
1 34815 0,0,1,8,1 home 3480
```

Pool 1 has two cartridges, 34813 and 34815.

## 2. Empty pool 1 by reassigning its cartridges to pool 5:

```
set scratch 5 348013 348015
```

## 3. Delete pool 1:

```
delete pool 1
```

**Notes**

None.

**See Also**

For information about...	See...
Cancelling a command	<a href="#">"cancel" on page 300</a>
Creating or modifying scratch pools	<a href="#">"define pool" on page 306</a>
Guidelines and procedures for managing scratch cartridges	<a href="#">"Managing Scratch Cartridges" on page 195</a>
Displaying scratch pool attributes	<a href="#">"query pool" on page 355</a>
Displaying the location and media type of a cartridge	<a href="#">"query cartridge" on page 362</a>
Setting or clearing cartridge scratch attributes	<a href="#">"set scratch" on page 372</a>

**Command Area Messages****Success Messages**

The following message appears when a `delete pool` command completes successfully:

```
Delete: Delete completed, Success.
```

In addition, the following message appears for each deleted pool.

Delete: Pool *pool\_id* deleted.

### Intermediate Messages

None

### Error Messages

Delete: Pool *pool\_id* failed, Pool not empty.

**Explanation:** ACSLS cannot delete the scratch pool because it is not empty.

**Variable:** *pool\_id* is the identifier of the requested pool.

## ■ **dismount**

The `dismount` command dismounts a cartridge from a transport.

### **Format**

```
dismount vol_id drive_id [force]
```

### **Options**

*vol\_id*  
specifies of the cartridge identifier.

*drive\_id*  
specifies the transport.

*force*  
forces a dismount of the actual cartridge in the specified transport, even if the *vol\_id* of the cartridge in the transport does not match the specified *vol\_id*.

This option also forces a dismount even if the transport is not unloaded.

### **Usage**

Use the `dismount` command to dismount a cartridge from a transport and place the cartridge in an available storage cell.

### **Dismount**

Use the `dismount` command without the *force* option to dismount a specified cartridge from a specified transport. A successful unforced dismount requires the following:

- The transport must be online
- The *vol\_id* of the cartridge in the transport must match the *vol\_id* you specify on the dismount command
- The transport must be unloaded

**Dismount force**

Use the `dismount` command with the `force` option to force a dismount of the actual cartridge mounted in a specified transport. The `vol_id` of the cartridge in the transport does not have to match the `vol_id` you specify on the `dismount` command. In addition, if the cartridge is not ready for dismounting, ACSLS forces the transport to automatically rewind, unload, and dismount the cartridge. The transport must be online.

You can use the `force` option to dismount a cartridge with an unreadable or unknown label or a cartridge that a client application did not dismount. Note that ACSLS returns the cartridge to an available storage cell, even if the label is unreadable or missing.

**CAUTION:**    **Dismounting a cartridge with dismount force will unconditionally abort any read/write activity between the drive and any application that is currently using the drive.**

This is due to the fact that a forced dismount immediately rewinds and unloads the cartridge in the transport specified.

**Examples**

To dismount cartridge EDU200 from transport 0,1,10,2:

```
dismount EDU200 0,1,10,2
```

**Notes**

None.

**See Also**

<b>For information about...</b>	<b>See...</b>
Cancelling a command	<a href="#">“cancel” on page 300</a>
Ejecting cartridges from the library	<a href="#">“eject” on page 314</a>
Making a CAP (manual mode) ready to enter labelled cartridges into the library	<a href="#">“enter” on page 318</a>
Mounting a scratch cartridge on a transport	<a href="#">“mount *” on page 330</a>
Mounting a data cartridge on a transport	<a href="#">“mount” on page 327</a>
Displaying CAP status	<a href="#">“query cap” on page 340</a>
Displaying transport status	<a href="#">“query drive” on page 343</a>
Setting CAP mode (manual or automatic)	<a href="#">“set cap mode” on page 365</a>
Setting CAP selection priority	<a href="#">“set cap priority” on page 367</a>

**Command Area Messages****Success Messages**

Dismount: *vol\_id* dismounted from *drive\_id*.

**Explanation:** An unforced dismount succeeded.

**Variable:**

- *vol\_id* is the identifier of the dismounted cartridge.
- *drive\_id* is the identifier of the specified transport.

Dismount: Forced dismount of *vol\_id* from *drive\_id*.

**Explanation:** A forced dismount succeeded.

**Variable:**

- *vol\_id* is the identifier of the dismounted cartridge.
- *drive\_id* is the identifier of the specified transport.

**Intermediate Messages**

None

**Error Messages**

Dismount: Dismount failed, ACS *acs\_id* full.

**Explanation:** ACSLS cannot dismount the cartridge because the ACS has no free storage cells. You must eject at least one other cartridge from the ACS so that ACSLS can dismount the cartridge from the transport.

**Variable:** *acs\_id* is the identifier of the ACS that contains the cartridge.

**Note:** Only online LSMs are used in locating free cells; therefore this message can occur even when there is an LSM with free storage cells if that LSM is offline.

Dismount: Dismount failed, Audit in progress.

**Explanation:** ACSLS cannot dismount the cartridge because an audit in progress has locked out access to the last unoccupied cell location in the ACS.

Dismount: Dismount failed, Cartridge in drive *drive\_id*, unreadable label

**Explanation:** ACSLS cannot dismount the cartridge because the cartridge either has no external label, an unreadable external label, or no virtual label. Use a forced dismount to dismount the cartridge.

**Variable:** *drive\_id* is the identifier of the specified transport.

Dismount: Dismount failed, Drive identifier *drive\_id* available.

**Explanation:** No cartridge is mounted in the specified transport.

**Variable:** *drive\_id* is the identifier of the specified transport.

Dismount: Dismount failed, Drive identifier *drive\_id* in use.

**Explanation:** ACSLS cannot dismount the cartridge because the cartridge is not rewound and unloaded. Either wait for the client application must either rewind and unload the cartridge, then enter a dismount command or use a forced dismount to dismount the cartridge.

**Variable:** *drive\_id* is the identifier of the specified transport.

Dismount: Dismount failed, Misplaced tape.

**Explanation:** ACSLS cannot dismount the cartridge because the external label of the cartridge does not match the cartridge identifier in the ACSLS database. The cartridge identifier in the ACSLS database is updated to match the external label of the cartridge. Retry the dismount.

Dismount: Dismount failed, Cartridge not in drive.

**Explanation:** ACSLS cannot dismount the cartridge because the external label of the cartridge does not match the cartridge identifier specified in the dismount. Reenter the dismount command with the correct cartridge identifier.

**Display Area Messages**

None.

**■ eject**

The `eject` command directs the robot to take cartridges from inside an LSM and place them into a CAP, where they can be removed by the operator.

If you select the `lsm_id` option, you can use multiple CAPs in a single LSM to eject cartridges.

**Format**

```
eject cap_id|lsm_id vol_id|volrange...
```

**Options**

`cap_id`

specifies the CAP used to eject the cartridges.

`lsm_id`

If you select the `lsm_id` option, you can use multiple CAPs that are available in a single LSM to eject cartridges. All CAPs that satisfy the requirements (such as manual, non zero priority, available) are unlocked for ejecting cartridges. You can eject cartridges through any or all the CAPs selected, and in any order. Eject initially fills the CAPs with cartridges in order of highest priority.

Example: You have two CAPs - one with priority 2 and one with priority 5; `eject` fills the CAP with priority 5 status first, then the CAP with priority 2 status. If there are only enough cartridges to be ejected for one CAP, then CAP with priority 5 is filled.

`vol_id | volrange`

specifies the external or virtual label types of the cartridge or range of cartridges to eject.

**Usage**

Use the `eject` command to eject cartridges from the library, which removes them from ACSLS control. The robot places the specified cartridge in the designated CAP, then ACSLS frees the cell locations where the cartridges were stored. Cartridge information is either retained if the `ABSENT_VOLUME_RETENTION_PERIOD` variable has a non-zero value, or deleted if the `ABSENT_VOLUME_RETENTION_PERIOD` is set to zero. If you specify more than a CAP full of cartridges on the `eject` command, empty the CAP when it fills, close the CAP, and continue the ejection until all cartridges are ejected.

You can use a single `eject` command to eject multiple cartridges not in a range by specifying multiple cartridge IDs separated by a space.

**Hint:** If your library has two connected LSMs and the PTP is down, to successfully eject cartridges, either:

- Specify a CAP in the LSM where the cartridge is stored. For example, if cartridge NN0100 is stored in LSM 0,0, to eject this cartridge through CAP 0,0,0:

```
eject 0,0,0 NN0101
```

- For the *cap\_id*, specify the LSM where the cartridge is stored but use a wildcard (\*) for the CAP number; ACSLS will select the highest priority CAP in the LSM. For example, if cartridge NN0114 is stored in LSM 0,0, to eject this cartridge through LSM 0,0's highest priority CAP:

```
eject 0,0,* NN0114
```

### **Examples**

To eject cartridge NN0101 through CAP 0,0,0:

```
eject 0,0,0 NN0101
```

To eject cartridge range NN0101-NN0109 through the highest priority CAP on ACS 0:

```
eject 0,* NN0101-NN0109
```

To eject cartridges NN0101, NN0103, NN0105, and NN0107 through the highest priority CAP on ACS 0:

```
eject 0,* NN0101 NN0103 NN0105 NN0107
```

To use multiple CAPs for ejecting cartridges in an CAP1,2:

```
eject 1,2 RB1000-RB2000
```

### **Notes**

Make sure you remove all ejected cartridges from the CAP. You must complete the ejection by unloading all ejected cartridges and closing the CAP door before you can use the CAP for another operation (such as an enter or audit).

**See Also**

<b>For information about...</b>	<b>See...</b>
Cancelling a command	<a href="#">“cancel” on page 300</a>
Dismounting a cartridge from a transport	<a href="#">“dismount” on page 310</a>
Guidelines and procedures for ejecting cartridges	<a href="#">“Ejecting Cartridges” on page 194</a>
Making a CAP (manual mode) ready to enter labelled cartridges into the library	<a href="#">“enter” on page 318</a>
Displaying CAP status	<a href="#">“query cap” on page 340</a>
Displaying transport status	<a href="#">“query drive” on page 343</a>
Displaying the location and media type of a cartridge	<a href="#">“query cartridge” on page 362</a>
Setting CAP mode (manual or automatic)	<a href="#">“set cap mode” on page 365</a>
Setting CAP selection priority	<a href="#">“set cap priority” on page 367</a>

**Command Area Messages****Success Messages**

Eject: Eject complete, *nn* cartridges ejected  
 Eject: *vol\_id* ejected from *cap\_id*

**Explanation:** ACSLS ejected the specified cartridges.

**Variable:**

- *nn* is the number of cartridges ejected.
- *vol\_id* is the cartridge identifier of the ejected cartridge.
- *cap\_id* is the CAP that contains the ejected cartridge.

**Intermediate Messages**

None

**Error Messages**

Eject: *vol\_id* Eject failed, CAP *cap\_id* full.

**Explanation:** A cartridge was not ejected because the CAP is full.

**Variable:**

- *vol\_id* is the cartridge identifier of the cartridge that was not ejected.

- *cap\_id* is the CAP specified for the ejection.

Eject: *vol\_id* Eject failed, CAP *cap\_id* in use.

**Explanation:** A cartridge was not ejected because the CAP is in use.

**Variable:**

- *vol\_id* is the cartridge identifier of the cartridge that was not ejected.
- *cap\_id* is the CAP specified for the ejection.

Eject: *vol\_id* Eject failed, Misplaced tape.

**Explanation:** ACSLS cannot eject the cartridge because the external label of the cartridge does not match the database's cartridge identifier for the storage cell. The database updates the cartridge identifier, media type, and cleaning cartridge attribute for the cartridge in the storage cell.

**Variable:** *vol\_id* is the cartridge identifier of the cartridge that was not ejected.

Eject: *vol\_id* Eject failed, Not in same ACS.

**Explanation:** A cartridge was not ejected because it is not in the ACS specified in the *cap\_id*.

**Variable:** *vol\_id* is the cartridge identifier of the cartridge that was not ejected.

Eject: *vol\_id* Eject failed, Cartridge identifier *vol\_id* not found.

**Explanation:** A cartridge was not ejected because it is not:

- In the storage cell specified in the database,
- In-transit, or
- In a transport.

ACSLs deletes the cartridge entry from the database.

**Variable:** *vol\_id* is the cartridge identifier that was deleted from the database.

Eject: *vol\_id* Eject failed, Cartridge in drive.

**Explanation:** A cartridge was not ejected because it is mounted in a transport.

**Variable:** *vol\_id* is the cartridge identifier of the cartridge that was not ejected.

Eject: *vol\_id* Eject failed, Cartridge *vol\_id* in use.

**Explanation:** A cartridge was not ejected because it is reserved for use by another request.

**Variable:** *vol\_id* is the cartridge identifier of the cartridge that was not ejected.

### ***Display Area Messages***

CAP *cap\_id* Remove cartridges from CAP.

**Explanation:** Either the CAP is full or all requested cartridges are in the CAP. Empty the CAP.

**Variable:** *cap\_id* is the CAP that contains the ejected cartridges.

CAP *cap\_id* Place magazines in CAP.

**Explanation:** The CAP requires magazines for the ejection. Open the CAP, place magazines inside, then close the CAP.

**Variable:** *cap\_id* is the CAP that requires magazines.

## ■ enter

This command allows you to set the CAP to operate in either manual or automatic mode. For procedures for setting the CAP to automatic mode or manual mode, refer to the section [“Entering Cartridges” on page 192](#).

- Automatic mode

When a CAP is in automatic mode, you can initiate an enter operation without issuing an enter command. You do this by opening the CAP door, placing one or more cartridges inside, and closing the CAP. The CAP is locked while the enter is being processed; when the enter operation is complete, the CAP is unlocked.

- Manual mode

When in manual mode, the CAP is locked and can only be used by issuing the following command prior to opening the CAP and entering the cartridges.

```
enter cap_id
```

### Options

*cap\_id*  
specifies the CAP.

*lsm\_id*  
If you select the *lsm\_id* option, you can use multiple CAPs that are available in a single LSM to enter cartridges. All CAPs that satisfy the requirements (such as manual, non zero priority, available) are unlocked for entering cartridges. You can enter cartridges through any or all the CAPs selected, and in any order. *venter* is not valid and returns a CAP invalid error.

### Usage

Use the `enter` command to make a manual mode CAP ready to enter labelled cartridges.

### Examples

To make CAP 0,0,2 ready to enter cartridges:

```
enter 0,0,2
```

To make the highest priority CAP in LSM 0,0 ready to enter cartridges:

```
enter 0,0,*
```

To use multiple CAPs for entering cartridges in an LSM 1,2:

```
enter 1,2
```

### Notes

Cartridges being entered into an LSM must be compatible with that LSM. For example, only T9840, T9940, and LTO cartridges can be entered into an L5500 LSM.

See Also

For information about...	See...
Cancelling a command	<a href="#">“cancel” on page 300</a>
Creating or modifying scratch pools	<a href="#">“define pool” on page 306</a>
Ejecting cartridges from the library	<a href="#">“eject” on page 314</a>

For information about...	See...
Guidelines and procedures for entering cartridges	<a href="#">“Entering Cartridges” on page 192</a>
Mounting a scratch cartridge on a transport	<a href="#">“mount *” on page 330</a>
Mounting a data cartridge on a transport	<a href="#">“mount” on page 327</a>
Displaying CAP status	<a href="#">“query cap” on page 340</a>
Setting CAP mode (manual or automatic)	<a href="#">“set cap mode” on page 365</a>
Setting CAP selection priority	<a href="#">“set cap priority” on page 367</a>
Making a CAP ready to enter unlabeled cartridges into the library	<a href="#">“venter” on page 388</a>

## Command Area Messages

### Success Messages

After all labelled cartridges have been entered, the following message appears, with the second line of the message repeated for each cartridge successfully entered:

```
Enter: Enter complete, nn cartridges entered
      Enter: vol_id Entered through cap_id
```

- *nn* is the total number of cartridges entered
- *vol\_id* is the cartridge identifier of the entered cartridge
- *cap\_id* is the CAP used to enter the cartridges

### Intermediate Messages

None.

### Error Messages

```
Enter: vol_id Enter failed, ACS acs_id full.
```

**Explanation:** A cartridge was not entered because the ACS has no free storage cells. You must eject at least one cartridge from the ACS so that ACSLS can enter the cartridge.

#### Variable:

- *vol\_id* is the external label of the cartridge that was not entered.
- *acs\_id* is the identifier of the ACS that has no free storage cells.

**Note:** ACSLS only searches online LSMs for free cells; this message can occur even if offline LSMS have free cells.

Enter: *vol\_id* Enter failed, Audit in progress.

**Explanation:** A cartridge was not entered because an audit has locked out access to a cell location required for the enter.

**Variable:** *vol\_id* is the external label of the cartridge that was not entered.

Enter: *vol\_id* Enter failed, CAP *cap\_id* in use.

**Explanation:** A cartridge was not entered because the specified CAP is in use for auditing, ejecting cartridges, or another enter process.

**Variable:**

*vol\_id* is the external label of the cartridge that was not entered.

*cap\_id* is the in-use CAP.

Enter: *vol\_id* Enter failed, Duplicate label.

**Explanation:** A cartridge was not entered because the cartridge identifier of a cartridge in the CAP already exists in the ACSLS database.

**Variable:** *vol\_id* is the external label of the cartridge that was not entered.

Enter: Enter failed, Unreadable label.

**Explanation:** A cartridge was not entered because the cartridge either has no external label or an unreadable external label.

Enter: *vol\_id* Enter failed, Unknown media type label.

**Explanation:** A cartridge was not entered because the cartridge does not have a media identifier on its external label.

**Variable:** *vol\_id* is the external label of the cartridge that was not entered.

### ***Display Area Messages***

CAP *cap\_id*: Place cartridges in CAP.

**Explanation:** The CAP is ready for cartridge entry. Open the CAP and enter cartridges.

**Variable:** *cap\_id* is the CAP used to enter cartridges.

CAP *cap\_id*: Remove cartridges from CAP.

**Explanation:** One or more cartridges cannot be entered.

**Variable:** *cap\_id* is the CAP used to enter cartridges. Open the CAP and remove the cartridges.

CAP *cap\_id*: CAP *cap\_id* Place magazines in CAP.

**Explanation:** The CAP uses magazines for cartridge entry. Load cartridges in the correct magazine, open the CAP, and insert the magazines.

**Variable:** *cap\_id* is the CAP used to enter cartridges.

## ■ idle

The `idle` command stops ACSLS from processing new requests.

### **Format**

`idle [force]`

**Hint:** You *must* enter the full command name for the `idle` command. ACSLS will reject all other forms of the command, such as `i`, `id`, or `idl`.

### **Options**

`force`

forces termination of new request processing.

### **Usage**

Use the `idle` command to stop ACSLS from processing new requests. For example, you idle ACSLS before doing maintenance or before terminating ACSLS.

### **Idle**

Entering an `idle` command without the force option puts ACSLS in the idle-pending state. ACSLS completes current and pending requests (except pending lock requests, which are cancelled) and rejects new requests except those listed in “Notes.” ACSLS then goes to idle state and does not process subsequent requests until restarted.

### **Idle force**

Entering an `idle` command with the force option puts ACSLS in the idle state. ACSLS cancels all current and pending requests and rejects new requests except those listed in “Notes.” ACSLS does not process subsequent requests until restarted. Because ACSLS does not complete current requests, forcing ACSLS to idle state may leave the database in an inconsistent state with the hardware, which requires an audit to correct.

**Hint:** Depending on the current requests processing when you enter an `idle force`, the event log may report process failures. If so, vary the affected LSMs offline and back online. See “Vary” on page 313 for information about the vary command.

**Examples**

To put ACSLS in idle-pending state:

```
idle
```

To force ACSLS to idle state:

```
idle force
```

**Notes**

In either idle or idle-pending state, ACSLS accepts new requests for the [cancel](#), [idle](#), [query](#), [start](#), and [vary](#) requests.

**See Also**

For information about...	See...
Displaying the status of a library component	<a href="#">“query commands” on page 337</a>
Starting ACSLS request processing	<a href="#">“start” on page 376</a>
Restarting ACSLS	<a href="#">“Starting ACSLS” on page 3</a>
Stopping ACSLS	<a href="#">“Stopping ACSLS” on page 3</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>

**Command Area Messages****Success Messages**

The following message appears when ACSLS request processing stops.

```
ACSLM Request Processing Stopped: Success.
```

**Intermediate Messages**

None

**Error Messages**

```
ACSLM Request Processing Stopped: status
```

**Explanation:** ACSLS did not stop processing requests.

**Variable:** status is the reason for the failure. For more information on common status messages, see ACSLS Messages.

**Display Area Messages****Variable:**

```
Server system idle
```

**Explanation:** Request processing has stopped, and ACSLS is in idle state.

Server system idle is pending

**Explanation:** ACSLS is processing current or pending requests; idle state is pending.

## ■ lock

The `lock` command locks a cartridge or transport to a lock ID that ACSLS assigns.

### Format

`lock type identifier...[wait]`

### Options

*type identifier*

specifies a library component. Table 24. lists the components that you can lock. You can specify either a transport or a cartridge but not both in a single Lock command. You can, however, use the same Lock ID for both transports and cartridges.

Table 26. Valid Components for Lock

Library Component	type	identifier
transport	<code>drive</code>	<code>drive_id</code>
cartridge	<code>cartridge</code>	<code>vol_id</code>

*wait*

specifies that the lock is pending if the component is unavailable (locked or in use). ACSLS then locks the component when it is available. You can clear or cancel a pending lock request. Idling ACSLS also cancels a pending lock request.

### Usage

Use the lock command to lock a cartridge or transport to a lock ID that ACSLS assigns. You can only lock available (not locked or in use) cartridges or transports.

**Note:** When you enter a lock command to lock a cartridge or transport, ACSLS assigns a lock ID to the cartridge or transport and then changes your lock ID to the lock ID of the cartridge or transport. You cannot use the set lock command to set your lock ID and then use the lock command to lock a cartridge or transport with the lock ID that you set with set lock.

**Examples**

To lock transport 0,1,10,2:

**lock drive 0,1,10,2**

To lock cartridge EDU445:

lock cartridge EDU445

**Notes**

None.

**See Also**

For information about...	See...
Removing all active or pending locks on a specified transport or cartridge	<a href="#">“clear lock” on page 304</a>
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Setting your lock ID	<a href="#">“set lock” on page 370</a>
Displaying your lock or user ID	<a href="#">“show” on page 375</a>
Removing active locks on transports or cartridges	<a href="#">“unlock” on page 380</a>

**Command Area Messages****Success Messages**

The following message appears when a lock request succeeds:

Lock: Lock completed, Success.

In addition, for each identifier in the request, one of the following messages appears:

Lock: Drive *drive\_id* locked under lock\_id *lock\_id*.  
 Lock: Cartridge *vol\_id* locked under lock\_id *lock\_id*.

Where:

- *drive\_id* is the locked transport.
- The *vol\_id* is the cartridge identifier of the locked cartridge.
- *lock\_id* is the lock ID.

**Intermediate Messages**

None.

**Error Messages**

One of the following messages appears if the lock request fails:

Lock: Lock of drive *drive\_id* failed, Drive in use.

**Explanation:** ACSLS cannot lock the specified transport because the transport is already locked or in use.

**Variable:** *drive\_id* is the transport that ACSLS cannot lock.

Lock: Lock of drive *drive\_id* failed, Lock failed.

**Explanation:** ACSLS cannot lock the specified transport. Reenter the lock command with the correct syntax and transport identifier.

**Variable:** *drive\_id* is the transport that ACSLS cannot lock.

Lock: Lock of cartridge *vol\_id* failed, Cartridge in use.

**Explanation:** ACSLS cannot lock the specified cartridge because the cartridge is already locked or in use.

**Variable:** *vol\_id* is the cartridge that ACSLS cannot lock.

Lock: Lock of drive *vol\_id* failed, Lock failed.

**Explanation:** ACSLS cannot lock the specified cartridge. Reenter the lock command with the correct syntax and cartridge identifier.

**Variable:** *vol\_id* is the cartridge that ACSLS cannot lock.

### ***Display Area Messages***

None.

## ■ **logoff**

The logoff command exits from a *cmd\_proc*.

### ***Format***

*logoff*

### ***Options***

None.

### ***Usage***

Use the logoff command to exit from *cmd\_proc*. If you are running an interactive (window) *cmd\_proc*, logoff also quits the *cmd\_proc* window.

### ***Examples***

To exit a *cmd\_proc*:

**logoff**

*Notes*

`logoff` exits from `cmd_proc` only, is valid in any ACSLS state, and does not affect ACSLS operations.

**See Also**

For information about...	See...
Starting a <code>cmd_proc</code>	<a href="#">“Starting a <code>cmd_proc</code>” on page 7</a>
Starting ACSLS	<a href="#">“Starting ACSLS” on page 3</a>
Stopping ACSLS	<a href="#">“Stopping ACSLS” on page 3</a>
Using a <code>cmd_proc</code>	<a href="#">“Using a <code>cmd_proc</code>” on page 4</a>

**Command Area Messages**

None.

**Display Area Messages**

None.

## ■ mount

The `mount` command mounts a data cartridge.

**Format**

```
mount vol_id drive_id [bypass] [readonly]
```

**Options**

`vol_id`  
specifies the cartridge.

`drive_id`  
specifies the transport.

`bypass`  
overrides some of the checks that ACSLS does before trying to mount a cartridge. For more information, see [“mount bypass” on page 328](#).

`readonly`  
specifies that the cartridge is mounted write-protected.

**CAUTION:** Some SCSI-attached libraries and some transports do not support mounts with write protect. Even if you specify the `readonly` option on a `mount` command, the transport may write to the cartridge. To write protect cartridges for these transports, you must use the cartridge's `readonly` protection (such as a thumbwheel).

**Usage**

Use the `mount` command to mount a data cartridge. You can only mount one cartridge on one transport each time you enter a `mount` command.

A successful mount requires the following:

- The cartridge and transport must be in the same ACS.
- The cartridge must be available and the transport must be online and available.

**mount bypass**

The `bypass` option overrides the checks that ACSLS does before trying to mount a cartridge as follows:

- The `bypass` option always overrides the ACSLS verification of the cartridge ID on the external label.
- The `bypass` option overrides the ACSLS verification of compatibility between the transport type and cartridge media. Use this option only when you know that the transport and cartridge media are, in fact, compatible. For example, you could use the `bypass` option to force a mount of a DD3C cartridge on an SD3 (Redwood) transport if the cartridge's external label is missing the 7th (media type) character.

**CAUTION:** Do not use the `bypass` option unless you know that the media type of the specified cartridge is compatible with the transport. Otherwise you can damage the cartridge, the transport, or both.

**Examples**

To mount cartridge EDU010 on transport 0,0,10,2:

```
mount EDU010 0,0,10,2
```

In the following example, YUMA15 is a DD3C cartridge whose cartridge label is missing the seventh character. The `bypass` option bypasses the media compatibility checks and forces a mount of the cartridge on transport 0,0,4,0, an SD3 transport.

To mount YUMA15 on transport 0,0,4,0 using the `bypass` option:

```
mount YUMA15 0,0,4,0 bypass
```

**Notes**

None.

**See Also**

<b>For information about...</b>	<b>See...</b>
Dismounting a cartridge from a transport	<a href="#">“dismount” on page 310</a>
Making a CAP (manual mode) ready to enter labelled cartridges into the library	<a href="#">“enter” on page 318</a>
Displaying CAP status	<a href="#">“query cap” on page 340</a>
Displaying transport status	<a href="#">“query drive” on page 343</a>
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Displaying the status of media-compatible transports for a specified data cartridge	<a href="#">“query mount” on page 350</a>
Displaying the location and media type of a cartridge	<a href="#">“query cartridge” on page 362</a>
Setting CAP mode (manual or automatic)	<a href="#">“set cap mode” on page 365</a>
Setting CAP selection priority	<a href="#">“set cap priority” on page 367</a>
Making a CAP ready to enter unlabeled cartridges into the library	<a href="#">“venter” on page 388</a>

**Command Area Messages****Success Messages**

Mount: *vol\_id* mounted on *drive\_id*

**Explanation:** ACSLS mounted the specified cartridge.

**Variable:**

- *vol\_id* is the cartridge identifier of the cartridge that ACSLS mounted.
- *drive\_id* is the transport that mounted the cartridge.

**Intermediate Messages**

None.

**Error Messages**

Mount: Mount failed, Audit in progress.

**Explanation:** ACSLS cannot mount the cartridge because an audit has locked out access to the cell location of the specified cartridge.

Mount: Mount failed, In use.

**Explanation:** ACSLS cannot mount the cartridge because either the transport is in use or the requested cartridge is reserved for another command.

Mount: Mount failed, Misplaced tape.

**Explanation:** ACSLS cannot mount the cartridge because the external label of the cartridge does not match the database's cartridge identifier for the storage cell. The database updates the cartridge identifier, media type, and cleaning cartridge attribute for the cartridge in the storage cell.

Mount: Mount failed, Not in same ACS.

**Explanation:** ACSLS cannot mount the cartridge because the specified cartridge and transport are not in the same ACS.

Mount: Mount failed, Cartridge in drive.

**Explanation:** ACSLS cannot mount the cartridge because the specified cartridge is already mounted in a transport.

Mount: Mount failed, Unreadable label.

**Explanation:** ACSLS cannot mount the cartridge because the cartridge has no label, an unreadable label, or no virtual label.

Mount: Mount failed, Invalid media type.

**Explanation:** ACSLS cannot mount the cartridge because the specified cartridge's media type is invalid.

Mount: Mount failed, Invalid drive type.

**Explanation:** ACSLS cannot mount the cartridge because the specified transport's type is invalid.

Mount: Mount failed, Incompatible media type.

**Explanation:** ACSLS cannot mount the cartridge because the cartridge's media type is incompatible with the specified transport.

### ***Display Area Messages***

None.

## ■ **mount \***

The `mount *` command selects a scratch cartridge and mounts it.

### ***Format***

`mount * drive_id [pool_id] [media media_type | media *]`

## Options

*drive\_id*  
specifies the transport.

*pool\_id*  
specifies the pool from which ACSLS selects the scratch cartridge.  
*pool\_id* is optional; if you do not specify a *pool\_id* ACSLS tries to find a scratch cartridge in the common pool (pool 0).

If you specify a *pool\_id* and the pool does not contain a scratch cartridge (or one of the correct media for mixed-media libraries) if the pool is set for overflow; ACSLS tries to find a scratch cartridge in the common pool (pool 0).

media *media\_type* | media \*  
specifies the cartridge media type. Specifying the media type is optional; see “Usage” for more information.

## Usage

Use the mount \* command to select a scratch cartridge and mount it. The following sections describe how ACSLS selects the scratch cartridge to be mounted and how to specify the cartridge media type for a scratch cartridge.

### How ACSLS Selects Scratch Cartridges

The mount \* command selects a scratch cartridge by:

- Creating a list of LSMs in the ACS, based on proximity to the LSM containing the specified transport.
- Examining each LSM in the list, until it finds scratch cartridges that meet the pool and media type criteria.
- Choosing the scratch cartridge with the least recent access date in that LSM.

The selected scratch cartridge is then mounted on the transport.

### Specifying the Media Type

You can specify cartridge media type in any of the following ways:

- Explicitly specify the media type in this form of the mount \* command:

mount \* *drive\_id* [*pool\_id*] *media media\_type*

Example: To mount a 9940 scratch cartridge from the common pool (pool 0):

```
mount * 0,0,10,2 media STK2P
```

- Use a wildcard (\*) for the media type so that ACSLS selects the media by scratch preferences; for more information see [“Setting Scratch Preferences” on page 172](#). To use scratch preferences, enter this form of the `mount` *Ⓜ* command:

```
mount * drive_id [pool_id] media *
```

Example: To use scratch preferences to mount a scratch cartridge from the common pool (pool 0):

```
mount * 0,0,10,2 media *
```

- Omit the media option in this form of the `mount` *Ⓜ* command. This instructs ACSLS to pick a media type compatible with the transport.

```
mount * drive_id [pool_id]
```

Example: To mount a scratch cartridge compatible with the transport from the common pool (pool 0):

```
mount * 0,0,10,2
```

### Examples

The following sections show `mount` scratch examples for single-media and mixed-media libraries.

#### Single-Media Libraries

To mount a scratch cartridge from pool 5 on transport 0,0,10,2:

```
mount * 0,0,10,2 5
```

**Hint:** If no cartridge is available from pool 5 and it has been set for `overflow` ACSLS will select a cartridge from the common pool (pool 0).

To mount a scratch cartridge from the common pool (pool 0) on transport 0,0,10,0

```
mount * 0,0,10,0
```

#### Mixed-Media Libraries

To mount a scratch cartridge from pool 5 with a media type of 3480 on transport 0,0,10,2:

```
mount * 0,0,10,2 5 media 3480
```

**Hint:** If no cartridge is available from pool 5 and it has been set for `overflow` ACSLS will select a cartridge with the specified media type from the common pool (pool 0).

To mount a scratch cartridge from pool 10 with media type determined by scratch preferences on transport 0,0,2,3:

```
mount * 0,0,2,3 10 media *
```

**Hint:** If no cartridge is available from pool 10 and it has been set for [overflow](#) ACSLS will select a cartridge with the specified media type from the common pool (pool 0).

To mount a scratch cartridge from the common pool (pool 0) with media type of 3480 on transport 0,0,10,2:

```
mount * 0,0,10,2 media 3480
```

To mount a scratch cartridge from the common pool (pool 0) with media type determined by scratch preferences on transport 0,0,2,3:

```
mount * 0,0,2,3 media *
```

To mount a scratch cartridge from the common pool (pool 0) with media compatible with transport 0,0,2,3:

```
mount * 0,0,2,3
```

### Notes

None.

### See Also

For information about...	See...
Creating or modifying scratch pools	<a href="#">“define pool” on page 306</a>
Deleting empty scratch pools	<a href="#">“delete pool” on page 308</a>
Dismounting a cartridge from a transport	<a href="#">“dismount” on page 310</a>
Managing scratch cartridges	<a href="#">“Managing Scratch Cartridges” on page 195</a>
Displaying transport status	<a href="#">“query drive” on page 343</a>
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Displaying the status of media-compatible transports for a specified scratch pool	<a href="#">“query mount **” on page 353</a>
Displaying scratch pool attributes	<a href="#">“query pool” on page 355</a>
Displaying scratch cartridge status	<a href="#">“query scratch” on page 359</a>
Displaying the location and media type of a cartridge	<a href="#">“query cartridge” on page 362</a>

For information about...	See...
Setting or clearing cartridge scratch attributes	<a href="#">“set scratch” on page 372</a>
Setting scratch preferences	<a href="#">“Setting Scratch Preferences” on page 172</a>

## Command Area Messages

### Success Messages

Mount: *vol\_id* mounted on *drive\_id*

**Explanation:** ACSLS mounted the specified cartridge.

**Variable:**

- *vol\_id* is the cartridge identifier of the cartridge that ACSLS mounted.
- *drive\_id* is the transport that mounted the cartridge.

### Intermediate Messages

None.

### Error Messages

Mount: Mount failed, Audit in progress.

**Explanation:** ACSLS cannot mount the cartridge because an audit has locked out access to the cell location of the specified cartridge.

Mount: Mount failed, In use.

**Explanation:** ACSLS cannot mount the cartridge because either the transport is in use or the requested cartridge is reserved for another command.

Mount: Mount failed, Misplaced tape.

**Explanation:** ACSLS cannot mount the cartridge because the external label of the cartridge does not match the database's cartridge identifier for the storage cell. The database updates the cartridge identifier, media type, and cleaning cartridge attribute for the cartridge in the storage cell.

Mount: Mount failed, Not in same ACS.

**Explanation:** ACSLS cannot mount the cartridge because the specified cartridge and transport are not in the same ACS.

Mount: Mount failed, Cartridge in drive.

**Explanation:** ACSLS cannot mount the cartridge because the specified cartridge is already mounted in a transport.

Mount: Mount failed, Unreadable label.

**Explanation:** ACSLS cannot mount the cartridge because the cartridge has no label, an unreadable label, or no virtual label.

Mount: Mount failed, Invalid media type.

**Explanation:** ACSLS cannot mount the cartridge because the specified cartridge's media type is invalid.

Mount: Mount failed, Invalid drive type.

**Explanation:** ACSLS cannot mount the cartridge because the specified transport's type is invalid.

Mount: Mount failed, Incompatible media type.

**Explanation:** ACSLS cannot mount the cartridge because the cartridge's media type is incompatible with the specified transport.

Mount: Mount failed, No compatible scratch cartridges in pool.

**Explanation:** ACSLS cannot mount the cartridge because there are no scratch cartridges in the ACS of the specified transport that match the scratch cartridge's media type. In addition, if the overflow attribute is set for the pool, there are no scratch cartridges with valid media types.

### ***Display Area Messages***

Pool *pool\_id*: low water mark warning.

**Explanation:** The number of cartridges in the specified scratch pool is less than or equal to the low cartridge threshold.

**Variable:** *low\_water\_mark* is the low threshold of the specified scratch pool.

Pool *pool\_id*: high water mark warning.

**Explanation:** The number of cartridges in the specified scratch pool is greater than or equal to the high cartridge threshold.

**Variable:** *high\_water\_mark* is the high water cartridge threshold of the specified scratch pool.

## ■ **move**

The `move` command moves a specified cartridge to an available storage cell in a specified LSM.

### ***Format***

```
move vol_id lsm_id
```

### **Options**

*vol\_id*  
specifies the cartridge.

*lsm\_id*  
specifies the LSM that will contain the moved cartridge.

### **Usage**

Use the `move` command to move a specified cartridge to an available storage cell in either:

- A different panel in the same LSM. For example, if you want to empty an entire panel in an LSM, you can move all the cartridges in that panel to a different locations in the same LSM.
- A different LSM.

You can move only one cartridge each time you enter the move command. If you specify the LSM where the cartridge currently resides, ACSLS will move the cartridge to another panel within that LSM. Otherwise, ACSLS will move it to the LSM you specify. You cannot cancel a `move` command.

A successful move requires the following:

- The cartridge must be available and in the same ACS as the specified LSM.
- The LSM where the cartridge currently resides and the specified LSM must both be online. The specified LSM must have at least one available storage cell. If the move is within the same LSM, there must be at least one available cell in a different panel in that LSM. If the move requires a pass-thru, any LSMs used must also be online.

### **Examples**

To move cartridge EDU010 (which resides in LSM 0,1) to a different panel in this LSM:

```
move EDU010 0,1
```

To move cartridge EDU010 (which resides in LSM 0,1) to LSM 0,2:

```
move EDU010 0,2
```

### **Notes**

None.

**See Also**

For information about...	See...
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Displaying the status of an LSM	<a href="#">“query lsm” on page 349</a>
Displaying the location and media type of a cartridge	<a href="#">“query cartridge” on page 362</a>

**Command Area Messages****Success Messages**

Move: *vol\_id* moved to location *cell\_id*

**Explanation:** ACSLS moved the specified cartridge to the specified cell location.

**Variable:**

- *vol\_id* is the cartridge identifier of the cartridge that ACSLS moved.
- *cell\_id* is the new cell location of the specified cartridge.

**Intermediate Messages**

None.

**Error Messages**

Move: Move failed,

**Explanation:**

**Variable:**

**Display Area Messages**

None.

## ■ query commands

The `query` commands display the status of a library component. See the following sections for more information about each query command, including format, options, and usage.

**Format**

The following shows the general format of the query commands:

```
query type [subtype | *] identifier... | al]
```

**Notes**

If a `cancel` request is issued against a pending or current query request, the display of information is halted.

**See Also**

For information about...	See...
Cancelling a command	<a href="#">“cancel” on page 300</a>
Display Command	<a href="#">“display Commands” on page 394</a>

**Command Area Messages****Success Messages**

**Note:** If the query succeeds, the requested status appears.

**Intermediate Messages**

None.

**Error Messages**

Library not available.

**Explanation:** The query failed because ACSLS is recovering (all queries *except* `query server`).

Cartridge identifier `vol_id` not found.

**Explanation:** The `query mount *command` cannot display status for the specified cartridge because it is not in the library.

**Variable:** `vol_id` is the specified cartridge.

Invalid media type

**Explanation:** The `query mount *` command cannot display status for the specified cartridge because it either has an invalid media type or you specified an invalid media type on the command.

**Variable:** `vol_id` is the specified cartridge.

**Display Area Messages**

None.

**■ query acs**

The `query acs` command displays ACS status.

**Format**

```
query acs acs_id... | all
```

**Options**`acs_id | all`

specifies the ACS to query or all for all ACSs.

**Usage**

Use the `query acs` command to display the status of an ACS in the following format:

<code>yyy-mm-ddhh:mm:ss</code>	ACS	Status	Free	Cell	Audit	Mount	Dismount	Enter	Eject
Identifier	State	Count	C/P						
<code>acs_id</code>	<code>state</code>	<code>count</code>	<code>n/n</code>						

**Where:**`acs_id`

is the ACS identifier.

`state` is one of the following ACS states:

`diagnostic`

The ACS will process only current and pending requests and will reject any new requests. The ACS is unavailable to client applications and can only be controlled via `cmd_proc`. Use the `vary` command to vary the ACS online.

`offline`

The ACS is offline.

`offline pending`

The ACS will process current and pending requests, then go offline. The ACS will reject any new requests.

`online`

The ACS is online.

`recovery`

The ACS is initializing or recovering from errors. Wait for the ACS to go online.

`count`

is the number of free cells in the ACS.

`n`

is the number of current (C) and pending (P) requests for the ACS for each command that requires library resources (audit, mount, dismount, enter, and eject).

**Examples**

To query ACS 1:

```
query acs 1
```

To query all ACSs in the library:

```
query acs all
```

**See Also**

For information about...	See...
Changing the state of a library component	<a href="#">“vary” on page 382</a>
Displaying request status	<a href="#">“query request” on page 358</a>

■ **query cap**

The `query cap` command displays CAP status.

**Format**

```
query cap cap_id... | all
```

**Options**

```
cap_id | all
```

specifies the CAP to query or all for all CAPs.

**Note:** You cannot specify a `cap_id` that contains asterisks.

**Usage**

Use the `query cap` command to display the status of a CAP.

The `query cap` command displays CAP status in the following format:

```

yyy-mm-ddhh:mm:ssCAP Status
IdentifierPrioritySize    State    Mode    Status
cap_idcap_prioritycap_size    cap_mode    status
    
```

Where:

`cap_id`  
is the CAP identifier.

*cap\_priority*

is the CAP priority.

*cap\_size*

is the number of cells in the CAP.

*cap\_state*

is one of the following CAP states:

online

The CAP is online.

offline

The CAP is offline.

offline-pending

The CAP will process current and pending requests, then go offline.  
The CAP will reject any new requests.

diagnostic

The CAP will process only current and pending requests and will reject any new requests. The CAP is unavailable to client applications and can only be controlled via `cmd_proc`. Use the `vary` command to vary the CAP online.

*recovery*

The CAP is initializing or recovering from errors. Wait for the CAP to go online.

*cap\_mode*

is one of the following CAP entry modes:

*manual*

You must unlock the CAP before entering cartridges.

*automatic*

The CAP is ready for cartridge entry.

*status*

is one of the following CAP statuses:

*available*

The CAP is available.

enter

The CAP is unavailable (reserved for entering cartridges).

eject

The CAP is unavailable (reserved for ejecting cartridges).

audit

The CAP is unavailable (reserved for audit processing).

**Examples**

To query CAP 0,1,0:

```
query cap 0,1,0
```

To query all CAPs in the library:

```
query cap all
```

**See Also**

For information about...	See...
Displaying request status	<a href="#">“query request” on page 358</a>
Setting CAP mode (manual or automatic)	<a href="#">“set cap mode” on page 365</a>
Setting CAP selection priority	<a href="#">“set cap priority” on page 367</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>

■ **query clean**

The `query clean` command displays cleaning cartridge status. Absent or ejected cartridges are not displayed.

**Format**

```
query clean vol_id... | all
```

**Options**

```
vol_id | all
```

specifies the cleaning cartridge to query or all for all cartridges.

**Usage**

Use the `query clean` command to display the status of a cleaning cartridge in the following format:

```

yyy-mm-ddhh:mm:ssCleaning Cartridge Status
IdentifierHome LocationMax UsageCurrent UseStatus Type
vol_idcell_idmax_usage current_use status type
    
```

Where:

*vol\_id*  
is the cartridge identifier of the cleaning cartridge.

*cell\_id*  
is the location of the cleaning cartridge.

*max\_usage*  
is the number of times the cleaning cartridge can be used.

*current\_usage*  
is the number of times the cleaning cartridge has been used.

*status*  
is the location of the cleaning cartridge:

home  
The cartridge is in a storage cell.

in drive  
The cartridge is in a transport.

in transit  
The cartridge is being moved.

*type* is the cartridge media type (for example, 3480, DD3D, DLTIII, or STK1R).

### **Examples**

To display status information for cleaning cartridge J35992:

```
query clean J35992
```

To query all cleaning cartridges:

```
query clean all
```

### **See Also**

For information about...	See...
Guidelines and procedures for cleaning transports	<a href="#">“Cleaning Cartridges” on page 199</a>
Setting cleaning cartridge attributes	<a href="#">“set clean” on page 369</a>

## ■ query drive

The `query drive` command displays transport status.

### **Format**

```
query drive drive_id... | all
```

**Options**

`drive_id | all`  
 specifies the transport to query or all for all transports.

**Usage**

Use the `query drive` command to display the status of a transport in the following format:

```

yyy-mm-ddhh:mm:ssDrive Status
IdentifierState Status      Cartridge  Type
drive_idstate  status      vol_id    type
  
```

**Where:**

`drive_id`  
 is the transport identifier.

The `state` is one of the following transport states:

`online`  
 The transport is online.

`offline`  
 The transport is offline.

`diagnostic`  
 The transport will process only current and pending requests and will reject any new requests. The transport is unavailable to client applications and can only be controlled via `cmd_proc`. Use the `vary` command to vary the transport online.

`recovery`  
 The transport is initializing or recovering from errors. Wait for the transport to go online.

`status`  
 is one of the following transport statuses:

`In use`  
 The transport has a cartridge mounted or is reserved for a mount.

**Possible Scenario:** You perform a `query drive all` and get a message saying the drive is in use. You then perform a `display drive *` and get a message saying the drive is reserved. This means: the reserved status for the drive implies that a mount request is in progress and that a cartridge is on its way to the drive. At the same time the drive is considered to be in use.

Available

The transport is available for a mount.

*vol\_id*

is the identifier of the cartridge in the transport. This field is blank if there is no cartridge in the transport or if the cartridge's external label is unreadable or unknown.

*drive\_type*

is the transport type.

### Examples

To query transport 0,3,1,0:

**query drive 0,3,1,0**

To query all transports:

**query drive all**

### See Also

For information about...	See...
Changing the state of a library component	<a href="#">“vary” on page 382</a>
Dismounting a cartridge from a transport	<a href="#">“dismount” on page 310</a>
Drive serial numbers	<a href="#">“display Commands” on page 394</a> and <a href="#">“display drive” on page 399</a>

## ■ query lmu

The `query lmu` command displays LMU and port status for both single-LMU and dual-LMU ACS configurations. If a library is partitioned, it also displays the partition ID.

**Note:** ACSLS supports dual-LMU configurations for only the 9330 LMU with host/LMU microcode compatibility Level 12. The same microcode level must be loaded in both LMUs.

### Format

```
query lmu acs_id... | all
```

### Options

```
acs_id | all
```

specifies the ACS whose LMUs you want to query or all to query LMUs for all ACSs.

**Usage**

Use the `query lmu` command to display LMU and port status for both single-LMU and dual-LMU ACS configurations in the following format:

```

ACS:  acsid  Mode:      mode      LMU Status
      Standby Status:      Master Status: status
                                     status

Port Port NamePort State   Role      Compat Level
port_id dev_name state    role(des) compat_level

```

**Where:**

*acs\_id*  
is the ACS identifier.

*mode*  
is the LMU mode (Dual LMU, Single LMU, or SCSI LMU).

*status*  
is the master or standby LMU status (Communicating, Communicating, or Offline). SCSI LMUs show - for status.

*port\_id*  
is the port identifier.

*dev\_name*  
is the port device name.

*state*  
is one of the following port states:

online  
The port is online.

offline  
The port is offline.

*role(des)*  
is the LMU's role and designation (A or B) where roles are:

Master  
LMU is in master role (the LMU is managing the ACS).

Standby  
LMU is in standby role (not managing the ACS, communicating with the master LMU and available for switchover).

**Note:** During switchover, the *role* field information is not current, and may display as a single dash (-). As the information becomes

current, ACSLS refreshes the *role* field with the actual role of each LMU.

`compat_level`

is the host/LMU microcode compatibility level. Level 11 or above is required for dual-LMU configurations.

### Examples

To display LMU and port status for all LMUs managing all ACSs:

```
query lmu all
```

To display LMU and port status for all LMUs managing ACSs 0 and 1:

```
query lmu 0 1
```

### See Also

For information about...	See...
Manually switching ACS management from the ACS's master LMU to the standby LMU	<a href="#">"switch lmu" on page 377</a>

## ■ query lock

The `query lock` command displays the lock status of a transport or cartridge.

### Format

```
query lock type identifier... | all
```

### Options

```
type identifier | all
```

specifies the transport or cartridge to query or all for all transports or cartridges as shown in [Table 27](#).

**Table 27. Valid Lock Types for query lock**

Library Component	<i>type</i>	<i>identifier</i>
transport	<code>drive</code>	<code>drive_id</code>
cartridge	<code>cartridge</code>	<code>vol_id</code>

**Usage**

Use the `query lock` command to display the lock status of a transport or cartridge in the following format:

```

yyy-mm-ddhh:mm:ssLock Status
IdentifierLock-idDuration Pending Status User Identifier

vol_idlock_id duration pending status user_id
or
drive_idlock_id duration pending status user_id

```

Where:

*vol\_id*  
is the identifier of the specified cartridge.

*drive\_id*  
is the identifier of the specified transport.

*lock\_id*  
is the lock ID.

*duration*  
is the amount of time, in seconds, that the lock has been active.

*pending*  
is the number of lock requests that are waiting for cartridge or transport.

*status* is one of the following statuses:

*available*  
The cartridge or transport is available.

*in use*  
The cartridge or transport is in use or is reserved for a mount.

*user\_id*  
is the user ID that has locked the cartridge or transport. The *user\_id* wraps after 80 characters.

**Examples**

To display lock status information for transport 1,0,4,0:

```
q loc dr 1,0,4,0
```

To display lock status information for all transports:

```
query lock drive all
```

To display lock status information for cartridge SL4493:

**query lock cartridge SL4493**

To display lock status information for all cartridges:

**query lock cartridge all**

### See Also

For information about...	See...
Removing all active or pending locks on a specified transport or cartridge	<a href="#">“clear lock” on page 304</a>
Locking transports and cartridges	<a href="#">“lock” on page 324</a>
Setting your lock ID	<a href="#">“set lock” on page 370</a>
Displaying your lock or user ID	<a href="#">“show” on page 375</a>
Removing active locks	<a href="#">“unlock” on page 380</a>

## ■ query lsm

The `query lsm` command displays LSM status.

### Format

```
query lsm lsm_id... | all
```

### Options

```
lsm_id | all
```

specifies the LSM to query or `all` for all locks.

Use the `query lsm` command to display the status of an LSM in the following format:

```

yyy-mm-ddhh:mm:ssLSM Status
IdentifierStateFree CellAudit Mount Dismount Enter Eject
Count C/P C/P C/P C/P C/P
lsm_idstate count n/n n/n n/n n/n n/n

```

Where:

```
lsm_id
```

is the LSM identifier.

`state` is one of the following LSM states:

*diagnostic*

The LSM will process only current and pending requests and will reject any new requests. The LSM is unavailable to client applications and can only be controlled via `cmd_proc`. Use the `vary` command to vary the LSM online.

*offline*

The LSM is offline.

*offline pending*

The LSM will process current and pending requests, then go offline. The LSM will reject any new requests.

*online*

The LSM is online.

*recovery*

The LSM is initializing or recovering from errors. Wait for the LSM to go online.

*count*

is the number of free storage cells in the LSM.

*n*

is the number of current (C) and pending (P) requests for the LSM for each command that requires library resources (`audit`, `mount`, `dismount`, `enter`, and `eject`).

**Examples**

To display status information for LSM 1 of ACS 0:

```
query lsm 0,1
```

To display status information for all LSMs:

```
query lsm all
```

**See Also**

For information about...	See...
Displaying request status	<a href="#">“query request” on page 358</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>

■ **query mount**

The `query mount` command displays the status of media-compatible transports for a specified data cartridge. These transports are not displayed if a cartridge(s) is absent or rejected.

**Format**

```
query mount vol_id
```

**Options**

*vol\_id* specifies the cartridge to query.

**Usage**

Use the `query mount` command to display the status of all library transports attached to the same ACS as the cartridge and compatible with the media type of the specified cartridge. `query mount` displays transport status for LSMs in any state (online, offline, offline-pending, or diagnostic).

The compatible transports are ordered by proximity to the specified cartridge in the following format:

```

yyy-mm-ddhh:mm:ssMount Status
IdentifierStatusDrive State      Status  Volume      Drive Type
vol_idvol_statdrive_id state    drive_statinu_id  drive_type

```

**Where:**

*vol\_id*  
is the identifier of the specified cartridge.

*vol\_stat*  
is the location of the cartridge:

home  
The cartridge is in a storage cell.

in drive  
The cartridge is in a transport.

in transit  
The cartridge is being moved or is missing.

*drive\_id*  
is a list of all library transports compatible with the media type of the specified cartridge.

*drive\_id*  
is the transport identifier.

*state*  
is one of the following transport states:

online

The transport is online.

offline

The transport is offline.

diagnostic

The transport will process only current and pending requests and will reject any new requests. The transport is unavailable to client applications and can only be controlled via `cmd_proc`. Use the vary command to vary the transport online.

recovery

The transport is initializing or recovering from errors. Wait for the transport to go online.

*status*

is one of the following transport statuses:

In use

The transport has a cartridge mounted or is reserved for a mount.

Available

The transport is available for a mount.

*inu\_id*

is the identifier of the cartridge in the transport. The cartridge id appears only if the *drive\_stat* is In use.

*drive\_type*

is the transport type.

### **Examples**

To display status information of transports ordered by proximity to cartridge ZUNI14:

```
query mount ZUNI14
```

**See Also**

For information about...	See...
Mounting a data cartridge on a transport	<a href="#">“mount” on page 327</a>
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Displaying the location and media type of a cartridge	<a href="#">“query cartridge” on page 362</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>

## ■ query mount \*

The `query mount *` command displays the status of media-compatible transports for a specified scratch pool (and, optionally, for a specific cartridge media type within the pool).

**Format**

```
query mount * pool_id... [media media_type | media *]
```

**Options**

`pool_id`  
specifies the scratch pool to query.

`media media_type | media *`  
specifies the media type.

**Usage**

Use the `query mount *` command to display the status of all library transports compatible with all cartridge media types in a specified scratch pool and in the same ACS as the cartridges. Pool 0 is the common scratch pool. Specify the `media_type` option to restrict the display to transports compatible with a specific cartridge media type within the pool. The displayed transports are ordered by proximity to the densest scratch pools. `query mount *` displays transport status for LSMs in any state (online, offline, offline-pending, or diagnostic).

The query `mount *` command displays transport status in the following format:

```

yyy-mm-ddhh:mm:ssMount Scratch Status
IdentifierDriveState Volume Status Drive Type
pool_iddrive_idstate vol_id drive_statdrive_type

```

Where:

*pool\_id*  
is the specified scratch pool.

*drive\_id*  
is a list of all library transports compatible with all media types of the specified pool (or a specific media type within the pool if a media type was specified).

*state*  
is one of the following transport states:

online  
The transport is online.

offline  
The transport is offline.

diagnostic  
The transport will process only current and pending requests and will reject any new requests. The transport is unavailable to client applications and can only be controlled via `cmd_proc`. Use the `vary` command to vary the transport online.

*recovery*  
The transport is initializing or recovering from errors. Wait for the transport to go online.

*vol\_id*  
is the identifier of the cartridge in the transport. The cartridge id appears only if the *drive\_stat* is In use.

*drive\_stat*  
is one of the following transport statuses:

In use  
The transport has a cartridge mounted or is reserved for a mount.

Available  
The transport is available for a mount.

*drive\_type*  
is the transport type.

### Examples

To display status of compatible transports listed by proximity to the largest concentration of scratch tapes in pool 5:

```
query mount * 5
```

To display status of compatible transports listed by proximity to the largest concentration of 3480 scratch tapes in common pool 0:

```
query mount * 0 media 3480
```

### See Also

For information about...	See...
Creating or modifying scratch pools	<a href="#">“define pool” on page 306</a>
Deleting empty scratch pools	<a href="#">“delete pool” on page 308</a>
Mounting a scratch cartridge on a transport	<a href="#">“mount *” on page 330</a>
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Displaying scratch pool attributes	<a href="#">“query pool” on page 355</a>
Displaying scratch cartridge status	<a href="#">“query scratch” on page 359</a>
Setting or clearing cartridge scratch attributes	<a href="#">“set scratch” on page 372</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>

## ■ query pool

The `query pool` command displays scratch pool attributes.

### Format

```
query pool pool_id... | all
```

### Options

```
pool_id | all
```

specifies the scratch pool to query or all for all pools. Pool 0 is the common pool.

**Usage**

Use the `query pool` command to display scratch pool attributes in the following format:

```

yyy-mm-dd hh:mm:ss Pool Status
Identifier Cartridge Count Low Water Mark High Water Mark Attributes
pool_id vol_count low_water_mark high_water_mark attribute

```

**Where:**

*pool\_id*  
is the specified scratch pool.

*vol\_count*  
is the number of scratch cartridges in the pool.

Absent and ejected scratch cartridges are not counted.

*low\_water\_mark*  
is the low cartridge warning threshold. If the scratch cartridge count falls below this threshold, ACSLS logs a warning message in the event log. A “-” after the value means that the scratch cartridge count is below the low cartridge threshold.

*high\_water\_mark*  
is the high cartridge warning threshold. If the scratch cartridge count reaches or exceeds this threshold, ACSLS logs a warning message in the event log. A “+” after the value means that the scratch cartridge count is at or above the high cartridge threshold.

*attribute*  
is displayed if `overflow` is set (via the `set scratch` command) for the specified scratch pool. `overflow` specifies that scratch cartridges are selected from the common scratch pool (Pool 0) if `mount scratch *` requests cannot be satisfied with a cartridge from the specified scratch pool.

**Examples**

To display status information for scratch pool 5:

```
query pool 5
```

To display status information for all scratch pools:

```
query pool all
```

**See Also**

For information about...	See...
Creating or modifying scratch pools	<a href="#">“define pool” on page 306</a>
Deleting empty scratch pools	<a href="#">“delete pool” on page 308</a>
Displaying scratch cartridge status	<a href="#">“query scratch” on page 359</a>
Setting or clearing cartridge scratch attributes	<a href="#">“set scratch” on page 372</a>

## ■ query port

The `query port` command displays port status.

**Format**

```
query port port_id... | all
```

**Options**

```
port_id | all
```

specifies the port to query or all for all ports.

**Usage**

Use the query port command to display the status of a port in the following format:

```
yyy-mm-dd hh:mm:ss Port Status
State Identifier

state port_id
```

Where:

```
state
```

is one of the following port states:

```
online
```

The port is online.

```
offline
```

The port is offline.

```
port_id
```

is the port identifier.

**Examples**

To display status information for port 0,0:

```
query port 0,0
```

To query all ports:

```
query port all
```

**See Also**

For information about...	See...
Changing the state of a library component	<a href="#">“vary” on page 382</a>

**■ query request**

The `query request` command displays request status.

**Format**

```
query request request_id... | all
```

**Options**

```
request_id | all
```

specifies the request to query or `all` for all requests.

**Usage**

Use the `query request` command to display the status of a request in the following format:

```

yyy-mm-dd hh:mm:ss Request Status
Identifier Command Status

request_id command                status

```

Where:

`request_id`  
is the ACSLS request identifier.

`command`  
is the ACSLS command that corresponds to the request identifier.

`status`  
is one of the following request statuses:

Current

ACSLs is processing the request.

Pending

The request is waiting to be processed.

Not found

The specified request is not a valid ACSLS request.

### **Examples**

To display status information for request 33179:

```
query request 33179
```

To display all current and pending requests:

```
query request all
```

### **See Also**

For information about...	See...
Canceling a command	<a href="#">“cancel” on page 300</a>

## ■ query scratch

The `query scratch` command displays the status of scratch cartridges in a pool sorted by ascending order of access date. The cartridges with the earliest access date are at the top of the list with the most recently used cartridges at the bottom of the list. Pool IDs in the first column do not appear in any particular order. Only those cartridges permitted through Access Control are displayed.

### **Format**

```
query scratch pool_id... | all
```

### **Options**

```
pool_id | all
```

specifies the scratch pool to query or all for all pools. Pool 0 is the common pool.

**Usage**

Use the `query scratch` command to display the status of scratch cartridges in a pool in the following format:

```

yyy-mm-ddhh:mm:ssScratch Status
Scratch PoolIdentifierHome LocationStatus      Type
pool_idvol_id cell_id      status            media_type

```

**Where:**

*pool\_id*  
is the specified scratch pool.

*vol\_id*  
is the identifier of the scratch cartridge.

*cell\_id*  
is the storage cell that contains the cartridge.

*status*  
is the location of the cartridge:

home  
The cartridge is in a storage cell.

in drive  
The cartridge is in a transport.

in transit  
The cartridge is being moved.

*media\_type*  
is the cartridge's media type (for example, 3480, 3490E, DD3D, or DLTIV).

**Note:** Scratch cartridges with a cartridge status of absent, ejected, or missing are not included in the `Query Scratch` output.

To see cartridges sorted by pool, you issue successive `Query Scratch` commands for each pool. Alternately, you can issue the `Display volume` command to see the information and sort it by pool, but you will not be guaranteed of getting the least recently used scratch cartridge.

**Examples**

To display status information for scratch cartridges in scratch pool 29015:

```
query scratch 29015
```

To display status information for scratch cartridges in all scratch pools:

**query scratch all**

### See Also

For information about...	See...
Creating or modifying scratch pools	<a href="#">“define pool” on page 306</a>
Deleting empty scratch pools	<a href="#">“delete pool” on page 308</a>
Setting or clearing cartridge scratch attributes	<a href="#">“set scratch” on page 372</a>

## ■ query server

The `query server` command displays ACSLS and library status.

### Format

`query server`

### Options

None.

### Usage

Use the `query server` command to display the status of ACSLS and the library in the following format:

```

yyy-mm-ddhh:mm:ssServer Status
IdentifierStateFree CellAudit Mount Dismount Enter Eject
Count C/P C/P C/P C/P C/P
state count n/n n/n n/n n/n n/n

```

Where:

Identifier  
is blank

state  
is one of the following ACSLS states:

idle  
ACSLS is idle (not processing requests).

idle pending

ACSLs will process current and pending request, reject new request, then go idle.

*recovery*

ACSLs is initializing (going to run state) or recovering from errors; ACSLS is not processing requests.

*run*

ACSLs is running (processing requests).

*count*

is the number of free storage cells in the library.

*n*

is the number of current (C) and pending (P) ACSLS requests for each command that requires library resources ([audit](#), [mount](#), [dismount](#), [enter](#), and [eject](#)).

**Hint:** You cannot cancel a `query server` request.

### Examples

To display status information on the server:

```
query server
```

### See Also

For information about...	See...
Changing the state of a library component	<a href="#">“vary” on page 382</a>
Displaying request status	<a href="#">“query request” on page 358</a>

## ■ query cartridge

The `query cartridge` command displays the location of a cartridge.

### Format

```
query cartridge vol_id... | all
```

### Options

```
vol_id | all
```

specifies the cartridge to query or `all` for all cartridges.

**Usage**

Use the `query cartridge` command to display the location of a cartridge in the following format:

```

yyy-mm-ddhh:mm:ssVolume Status
IdentifierStatusCurrent Location Type

vol_idstatus location media_type

```

**Where:**

`vol_id`  
is the cartridge identifier.

`status`  
is the location of the cartridge:

`home`  
The cartridge is in a storage cell.

`in drive`  
The cartridge is in a transport.

`in transit`  
The cartridge is being moved or is missing.

`absent`  
The cartridge cannot be found.

`ejected`  
The cartridge has been ejected from the library.

`location`  
specifies the location as one of the following:

- If the status is `home`, the location is a storage cell identifier.
- If the status is `in transit`, the location is either a cell identifier or a transport identifier.
- If the status is `in drive`, the location is a transport identifier.

`media_type`  
is the cartridge's media type (for example, 3480, 3490E, DD3D, or DLTIV).

**Examples**

To display status information for cartridge 2903B:

```
query cartridge 2903B
```

To display all cartridges in the library:

**query cartridge all**

### See Also

For information about...	See...
Other cartridge information	<a href="#">“Creating a Custom Volume Report” on page 233</a>
Display Command	<a href="#">“display Commands” on page 394</a> and <a href="#">“display volume” on page 409</a>
Mounting a data cartridge on a transport	<a href="#">“mount” on page 327</a>
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Displaying the status of media-compatible transports for a specified data cartridge	<a href="#">“query mount” on page 350</a>
Displaying scratch pool attributes	<a href="#">“query pool” on page 355</a>
Setting or clearing cartridge scratch attributes	<a href="#">“set scratch” on page 372</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>

## ■ set commands

The `set` commands set various attributes of different library components. See the following sections for more information about each `set` command, including format, options, usage, and messages.

### Format

The following shows the general format of the set commands:

```
set type [off | subtype] [*] identifier...
```

### See Also

For information about...	See...
Cancelling a command	<a href="#">“cancel” on page 300</a>

### Command Area Messages

#### Success Messages

The following message appears when a set command succeeds:

Set: Set completed, Success.

See each [set](#) command for specific success messages.

### **Intermediate Messages**

None.

### **Error Messages**

See each [set](#) command for specific messages.

### **Display Area Messages**

See each [set](#) command for specific messages.

## ■ set cap mode

The [set cap mode](#) command sets a CAP's entry mode.

### **Format**

```
set cap mode cap_mode cap_id
```

### **Options**

*cap\_mode*

specifies manual or automatic CAP mode.

*cap\_id*

specifies the CAP identifier.

You cannot specify a CAP identifier that contains an asterisk (\*) unless the priority is set.

### **Usage**

Use the [set cap mode](#) command to set a CAP's entry mode to one of the following modes:

manual

You must enter an enter command before entering cartridges.

automatic

You can enter cartridges without first entering an enter command

**Note:** You cannot set the CAP mode to automatic in a partitioned library.

**Hint:** You cannot change a CAP's mode while the CAP is in use. That is, if the door is open during either manual or automatic enter operations, you cannot change its mode until you complete the enter operation.

### **Examples**

To set CAP 0,3,1 in manual mode:

```
set cap mode manual 0,3,1
```

To set CAP 0,3,1 in automatic mode:

```
set cap mode automatic 0,3,1
```

**See Also**

For information about...	See...
Making a CAP (manual mode) ready to enter labelled cartridges into the library	<a href="#">“enter” on page 318</a>
Displaying CAP status	<a href="#">“query cap” on page 340</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>
Making a CAP ready to enter unlabeled cartridges into the library	<a href="#">“venter” on page 388</a>

**Command Area Messages**

**Success Messages**

Set: CAP *cap\_id*, mode changed to *cap\_mode*.

**Explanation:** ACSLS changed the mode of the specified CAP.

**Variable:**

- *cap\_id* is the CAP whose mode is changed.
- *cap\_mode* is the new enter mode of the CAP.

**Intermediate Messages**

None.

**Error Messages**

Set: CAP *cap\_id* Set failed, Incorrect attribute.

**Explanation:** ACSLS cannot change the mode of the specified CAP because you specified an invalid CAP mode.

**Variable:** *cap\_id* is the CAP whose mode is not changed.

CAP *cap\_id*: Automatic mode.

**Explanation:** ACSLS changed the specified CAP's mode to automatic.

**Variable:** *cap\_id* is the CAP whose mode is changed.

CAP *cap\_id*: Manual mode.

**Explanation:** ACSLS changed the specified CAP's mode to manual.

**Variable:** *cap\_id* is the CAP whose mode is changed.

## ■ set cap priority

The `set cap priority` command sets a CAP's automatic selection priority.

### **Format**

```
set cap priority cap_priority cap_id
```

### **Options**

*cap\_priority*

specifies the CAP priority. Valid values are 0 to 16, where 16 is the highest priority. All CAPs initially have a 0 priority, which means that ACSLS does not automatically select the CAP.

*cap\_id*

specifies the CAP identifier. You must specify a particular CAP; you cannot specify an asterisk (\*) to set the same priority for all CAPs.

### **Usage**

Use the `set cap priority` command to set a CAP's automatic selection priority.

If a CAP request specifies an asterisk (\*) for the CAP ID, ACSLS automatically selects an available CAP with highest non-zero priority for each ACS specified in the request.

### **Examples**

To assign priority 16 to CAP 0,3,1:

```
set cap priority 16 0,3,1
```

**See Also**

For information about...	See...
Updating the ACSLS database to match the actual inventory of library cartridges	<a href="#">“audit” on page 294</a>
Ejecting cartridges from the library	<a href="#">“eject” on page 314</a>
Making a CAP (manual mode) ready to enter labelled cartridges into the library	<a href="#">“enter” on page 318</a>
Displaying CAP status	<a href="#">“query cap” on page 340</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>
Making a CAP ready to enter unlabeled cartridges into the library	<a href="#">“venter” on page 388</a>

**Command Area Messages**

Updates the ACSLS database to match the actual inventory of library cartridges.

**Success Messages**

Set: CAP *cap\_id*, priority changed to *cap\_priority*.

**Explanation:** ACSLS changed the priority of the specified CAP.

**Variable:**

- *cap\_id* is the CAP whose priority is changed.
- *cap\_priority* is the new CAP priority.

**Intermediate Messages**

None.

**Error Messages**

Set: CAP *cap\_id* Set failed, Incorrect attribute.

**Explanation:** ACSLS cannot change the mode of the specified CAP because you specified an invalid CAP priority.

**Variable:** *cap\_id* is the CAP whose priority is not changed.

**Display Area Messages**

None.

## ■ set clean

The `set clean` command sets cleaning cartridge attributes.

### Format

```
set clean max_usage | off vol_id | volrange
```

### Options

```
max_usage | off
```

specifies the number of times a cleaning cartridge is used before ACSLS stops selecting the cartridge to clean transports. `off` specifies that ACSLS will not select the cartridge and redefines the cartridge as a data cartridge.

```
vol_id | volrange
```

specifies the cleaning cartridge or range of cartridges.

### Usage

Use the `set clean` command to set the number of times ACSLS will select a cleaning cartridge. You also use `set clean` to set a cartridge's cleaning cartridge attribute off, which you do if you have incorrectly defined a data cartridge as a cleaning cartridge.

### Examples

To set the maximum use to 10 for cleaning cartridges CLN108 - CLN112:

```
set clean 10 CLN108-CLN112
```

To set the cleaning cartridge attribute off and redefine cartridges HRR234 - HRR244 as data cartridges:

```
set clean off HRR234-HRR244
```

### See Also

For information about...	See...
Guidelines and procedures for cleaning transports	<a href="#">“Cleaning Cartridges” on page 199</a>
Displaying cleaning cartridge status	<a href="#">“query clean” on page 342</a>

### Command Area Messages

#### Success Messages

```
Set: cartridge vol_id is a cleaning cartridge.
```

**Explanation:** ACSLS specified that the cleaning cartridge is valid for transport cleaning.

**Variable:** `vol_id` is the valid cleaning cartridge.

Set: cartridge *vol\_id* is not a cleaning cartridge.

**Explanation:** ACSLS specified that the cleaning cartridge is not valid for transport cleaning.

**Variable:** *vol\_id* is the invalid cleaning cartridge.

#### Intermediate Messages

None.

#### Error Messages

Set: Clean *vol\_id* Set failed, Incorrect attribute.

**Explanation:** ACSLS cannot change the cleaning attribute because the specified cartridge is not a cleaning cartridge.

**Variable:** *vol\_id* is the data or scratch cartridge.

#### Display Area Messages

None.

## ■ set lock

The `set lock` command sets your lock ID.

#### Format

```
set lock lock_id
```

#### Options

*lock\_id*

specifies the lock ID. Valid lock IDs are 0 to 32767.

#### Usage

Use the `set lock` command to set or change your lock ID. Change your current lock ID when you want to remove locks on a transport or cartridge whose lock ID does not match your current lock ID as follows:

- To clear all locks on a specified transport or cartridge, set your lock ID to 0, then enter a `clear lock` command.
- To remove active locks on transports or cartridges, set your lock ID to the lock ID of the locked components, then enter an unlock command.

**Note:** When you enter a `lock` command to lock a cartridge or transport, ACSLS assigns a lock ID to the cartridge or transport and then changes your lock ID to the lock ID of the cartridge or transport. You cannot use the `set lock` command to set your lock ID and then use

the `lock` command to lock a cartridge or transport with the lock ID that you set with `set lock`.

### Examples

To change your current lock ID `lock` to new lock ID 354:

```
set lock 354
```

### See Also

For information about...	See...
Removing all active or pending locks on a specified transport or cartridge	<a href="#">“clear lock” on page 304</a>
Locking transports and cartridges	<a href="#">“lock” on page 324</a>
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Displaying your lock or user ID	<a href="#">“show” on page 375</a>
Removing active locks	<a href="#">“unlock” on page 380</a>

### Command Area Messages

#### Success Messages

```
Set: Changed lock identifier from previous_lock_id to lock_id.
```

**Explanation:** ACSLS changed your lock ID.

**Variable:**

*previous\_lock\_id* was your previous lock ID.

*lock\_id* is your new lock ID.

#### Intermediate Messages

None.

#### Error Messages

None.

#### Display Area Messages

None.

## ■ set owner

The `set owner` command sets cartridge ownership.

### **Format**

```
set owner owner_id cartridge vol_id | volrange
```

### **Options**

`owner_id`

specifies the owner identifier. You must enclose this value in quotes ("").

`cartridge vol_id | volrange`

specifies the cartridge or range of cartridges.

### **Usage**

Use the `set owner` command to set cartridge ownership. You must enter `set owner` from `cmd_proc`; you cannot set cartridge ownership from a client application.

### **Examples**

To assign ownership of cartridge YUMA06:

```
set owner "cray" cartridge YUMA06
```

### **Notes**

None.

### **See Also**

For information about...	See...
cartridge access control	<a href="#">"How ACSLS Access Control Works" on page 135</a>

### **Command Area Messages**

None.

### **Display Area Messages**

None.

## ■ set scratch

The `set scratch` command sets or clears a cartridge's scratch attribute and assigns the cartridge to a pool.

### **Format**

```
set scratch [off] pool_id vol_id | volrange
```

### Options

*off*

specifies that the cartridge is a data cartridge.

*pool\_id*

specifies the cartridge's scratch pool. Specify an asterisk (\*) to reassign a data cartridge to its current pool.

*vol\_id* | *volrange*

specifies the cartridge or range of cartridges.

### Usage

Use the set scratch command to set or clear a cartridge's scratch attribute and assign the cartridge to a pool.

### Examples

To define cartridges YUMA10-YUMA20 as scratch cartridges and assign them to scratch pool 5:

```
set scratch 5 YUMA10-YUMA20
```

To move scratch cartridges YUMA10-YUMA15 to pool 10:

```
set scratch 10 YUMA10-YUMA15
```

To “unscratch” (change from scratch to data) cartridges YUMA16-YUMA20 and move them to the common pool (pool 0):

```
set scratch off 0 YUMA16-YUMA20
```

To scratch to data cartridges YUMA16-YUMA20 and keep them in their current pool:

```
set scratch * YUMA16-YUMA20
```

### See Also

For information about...	See...
Displaying scratch cartridge status	<a href="#">“query scratch” on page 359</a>
Displaying scratch pool attributes	<a href="#">“query pool” on page 355</a>
Creating or modifying scratch pools	<a href="#">“define pool” on page 306</a>
Deleting empty scratch pools	<a href="#">“delete pool” on page 308</a>

## **Command Area Messages**

### **Success Messages**

Set: cartridge *vol\_id* in tape pool *pool\_id* is a scratch cartridge.

**Explanation:** ACSLS changed the specified data cartridge to a scratch cartridge and assigned it to a pool.

**Variable:**

- *vol\_id* is the specified cartridge.
- *pool\_id* is the pool to which the cartridge is assigned.

Set: cartridge *vol\_id* in tape pool *pool\_id* is a data cartridge.

**Explanation:** ACSLS changed the specified scratch cartridge to a data cartridge.

**Variable:** *vol\_id* is the specified cartridge.

### **Intermediate Messages**

None.

### **Error Messages**

Set: Scratch *vol\_id* Set failed, Incorrect attribute.

**Explanation:** ACSLS cannot change the scratch attribute because the specified cartridge is a cleaning cartridge.

**Variable:** *vol\_id* is the cleaning cartridge.

### **Display Area Messages**

Pool *pool\_id*: low water mark warning.

**Explanation:** The number of cartridges in the specified scratch pool is less than or equal to the low threshold.

**Variable:** *low\_water\_mark* is the low threshold of the specified scratch pool.

Pool *pool\_id*: high water mark warning.

**Explanation:** The number of cartridges in the specified scratch pool is greater than or equal to the high threshold.

**Variable:** *high\_water\_mark* is the high threshold of the specified scratch pool.

## ■ show

The `show` command displays your lock ID or user ID.

### **Format**

`show type`

### **Options**

*type*

specifies one of the following types:

`lock`

Your lock ID.

`user`

Your user ID.

### **Usage**

Use the `show` command to display your lock ID or user ID.

### **Examples**

To display requestor's `user_id`:

```
show user
```

To display current `lock_id`:

```
show lock
```

### **Notes**

None.

### **See Also**

For information about...	See...
Removing all active or pending locks on a specified transport or cartridge	<a href="#">“clear lock” on page 304</a>
Locking transports and cartridges	<a href="#">“lock” on page 324</a>
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Setting your lock ID	<a href="#">“set lock” on page 370</a>
Removing active locks	<a href="#">“unlock” on page 380</a>

### ***Command Area Messages***

#### **Success Messages**

Show: Current lock identifier is *lock\_id*.

**Explanation:** Your current lock ID is *lock\_id*.

Show: User identifier is *user\_id*.

**Explanation:** Your current user ID is *user\_id*.

#### **Intermediate Messages**

None.

#### **Error Messages**

None.

### ***Display Area Messages***

None.

## ■ **start**

The start command starts ACSLS request processing.

#### ***Format***

start

#### ***Options***

None.

#### ***Usage***

Use the start command to put ACSLS in run state and start ACSLS request processing. You typically use the start command to restart request processing if ACSLS is idled.

#### ***Examples***

To restart ACSLS request processing:

**start**

If ACSLS is in run state, entering the start command has no effect.

**See Also**

For information about...	See...
Stopping ACSLS from processing new requests	<a href="#">“idle” on page 322</a>
Displaying the status of a library component	<a href="#">“query commands” on page 337</a>
Restarting ACSLS	<a href="#">“Starting ACSLS” on page 3</a>
Stopping ACSLS	<a href="#">“Stopping ACSLS” on page 3</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>

**Command Area Messages****Success Messages**

ACSLM Request Processing Started: Success.

**Explanation:** ACSLS started processing requests.

**Intermediate Messages**

None.

**Error Messages**

ACSLM Request Processing Not Started: *status*

**Explanation:** ACSLS did not start processing requests.

**Variable:** *status* is the reason for the failure. For more information on common status messages, see ACSLS Messages.

**Display Area Messages**

Server system running

**Explanation:** ACSLS started processing requests.

## ■ switch lmu

In dual-LMU configurations, the `switch lmu` command manually switches ACS management from the ACS's master LMU to the standby LMU.

**Note:** ACSLS supports dual-LMU configurations for only the 9330 LMU with host/LMU microcode compatibility Level 11 or above loaded. The same microcode level must be loaded in both LMUs.

**Format**

```
switch lmu acs_id
```

**Options***acs\_id*

specifies the ACS to switch ACS management from the master to the standby LMU.

**Usage**

Use the `switch lmu` command to manually switch ACS management from the ACS's master LMU to the standby LMU. Before you enter a `switch lmu` command, make sure that:

- ACSLS is in run state
- The ACS you specify is in online or diagnostic state
- At least one port is online to each LMU

**Examples**

In dual-LMU configurations, you can use the `switch lmu` command to switch ACS management from the ACS's master LMU to the standby LMU. Assume the following dual-LMU configuration:

- LMU A is in the master role and LMU B is in the standby role.
- Both LMUs manage ACS 0.
- The ports to LMU A have port IDs 0,0 and 0,1.

The following procedure tells how to use the `switch lmu` command to let you replace a LAN cable between the LMUs.

**To replace a LAN cable between dual LMUs configured as described above, do the following:**

1. Switch to LMU B:
 

```
switch lmu 0
```
2. Vary the ports to LMU A offline:
 

```
vary port 0,0 0,1 offline
```
3. Replace the LAN cable.
4. Vary the ports to LMU A back online:
 

```
vary port 0,0 0,1 online
```
5. Switch back to LMU A:
 

```
switch lmu 0
```

**See Also**

For information about...	See...
Displaying ACS status	<a href="#">“query acs” on page 338</a>
Displaying LMU status	<a href="#">“query lmu” on page 345</a>
Displaying port status	<a href="#">“query port” on page 357</a>
Displaying ACSLS and library status	<a href="#">“query server” on page 361</a>
Starting ACSLS request processing	<a href="#">“start” on page 376</a>
Restarting ACSLS	<a href="#">“Starting ACSLS” on page 3</a>
Changing the state of a library component	<a href="#">“vary” on page 382</a>

**Command Area Messages****Success Messages**

Switch: Switch lmu completed for *acs\_id*, Success.

**Explanation:** ACSLS switched ACS management from the master to the standby LMU.

**Variable:** *acs\_id* is the ACS whose LMUs switched roles.

Switch: Switch lmu initiated for *acs\_id*, Success.

**Explanation:** ACSLS started switching ACS management from the master to the standby LMU.

**Variable:** *acs\_id* is the ACS whose LMUs are switching roles.

Switch: Switch lmu failed, *acs\_id* not found.

**Explanation:** You specified an invalid ACS.

**Variable:** *acs\_id* is the ACS identifier you specified on the `switch lmu` command.

Switch: Switch lmu failed, *acs\_id* is offline.

**Explanation:** You specified an ACS that is offline. Use the vary command to vary the ACS to online or diagnostic mode.

**Variable:** *acs\_id* is the ACS identifier you specified on the `switch lmu` command.

Switch: Switch lmu failed, *acs\_id* not configured for dual lmu.

**Explanation:** You specified an ACS that is not configured with master and standby LMUs. Either:

- Reconfigure the specified ACS, then reenter the `switch lmu` command
- Enter a `switch lmu` command that specifies an LMU that is configured with master and standby LMUs.

**Variable:** *acs\_id* is the ACS identifier you specified on the `switch lmu` command.

Switch: Switch lmu failed, not communicating.

**Explanation:** An LMU switch failed because the standby LMU is not communicating. Check the standby LMU for communications problems.

Switch: Switch lmu failed, switch already active.

**Explanation:** An LMU switch is in progress for the ACS you specified. Wait until the switch completes, then, if desired, reenter the `switch lmu` command.

Switch: Switch lmu failed, port is offline.

**Explanation:** You specified an ACS with no LMU ports online Vary at least one port online to each LMU.

### Display Area Messages

None.

## ■ unlock

The `unlock` command removes active locks (associated with your current lock ID) on a specified transport or cartridge or removes all active locks.

### Format

`unlock type identifier...| all`

### Options

*type identifier*

specifies a library component. [Table 28](#) lists the components that you can unlock.

**Table 28. Valid Components for Unlock**

Library Component	type	identifier
transport	<i>drive</i>	<i>drive_id</i>

**Table 28. Valid Components for Unlock**

Library Component	type	identifier
cartridge	<code>cartridge</code>	<code>vol_id</code>

`all`  
specifies all active locks.

### Usage

Use the `unlock` command to remove active locks on specified cartridges and transports or remove all active locks. The lock ID of the component must match your current lock ID.

When you unlock a component that is not in use, ACSLS checks for pending locks on the component. If there is a pending lock, ACSLS locks the component with the lock ID of the pending lock.

**Note:** The `unlock` command always resets your lock ID to 0.

### Examples

To remove active locks on transport 0,0,2,0:

```
unlock drive 0,0,2,0
```

To remove active locks on all locked cartridges:

```
unlock cartridge all
```

### See Also

For information about...	See...
Removing all active or pending locks on a specified transport or cartridge	<a href="#">“clear lock” on page 304</a>
Locking transports and cartridges	<a href="#">“lock” on page 324</a>
Displaying the lock status of a transport or cartridge	<a href="#">“query lock” on page 347</a>
Setting your lock ID	<a href="#">“set lock” on page 370</a>
Displaying your lock or user ID	<a href="#">“show” on page 375</a>

### Command Area Messages

#### Success Messages

```
Unlock: Unlock completed, Success.  
Unlock: drive drive_id unlocked.
```

**Explanation:** ACSLS unlocked the specified transport.

**Variable:** `drive_id` is the unlocked transport.

Unlock: Unlock completed, Success.  
 Unlock: cartridge *vol\_id* unlocked.

**Explanation:** ACSLS unlocked the specified cartridge.

**Variable:** *vol\_id* is the unlocked cartridge.

**Intermediate Messages**

None.

**Error Messages**

Unlock: Unlock of transport *drive\_id* failed, *status*.

**Explanation:** ACSLS cannot unlock the transport.

**Variable:** *status* is the reason for the failure. For more information on common status messages, see ACSLS Messages.

Unlock: Unlock of cartridge *vol\_id* failed, *status*.

**Explanation:** ACSLS cannot unlock the cartridge.

**Variable:** *status* is the reason for the failure. For more information on common status messages, see ACSLS Messages.

**Display Area Messages**

None.

■ vary

The vary command changes the state of a library component.

**Format**

*v*ary *type identifier...state* [*f*orce]

**Options**

*type* [*identifier*]

specifies a library component. [Table 29](#) lists the components you can vary.

**Table 29. Valid Components for Vary**

ACS Component	type	identifier
ACS	<i>acs</i>	<i>acs_id</i>
LSM	<i>lsm</i>	<i>acs_id,lsm_id</i>
CAP	<i>cap</i>	<i>cap_id</i>
transport	<i>drive</i>	<i>drive_id</i>
port	<i>port</i>	<i>port_id</i>

*state*

specifies one of the following device states: `online` (online to clients and `cmd_proc`), `offline`, or `diagnostic` (online to `cmd_proc` only).

You can vary an ACS, LSM, CAP, or transport online, offline, or diagnostic. You can vary a port either online or offline. See “Usage” for more information about device states.

*force*

varies an ACS, LSM, CAP, or transport `offline` after processing only the current robotic request.

## Usage

Use the `vary` command to change the state of an ACS, LSM, CAP, transport, or port. The following sections tell how each of the device states you can specify affects library components.

### vary offline

Table 30 shows the results of `vary offline` for each ACS component. When you vary a component offline and it first goes to offline-pending state, ACSLS processes all active and pending requests for the component, rejects new requests, then varies the component offline.

**Table 30. Vary Offline Results**

ACS Component	Results
ACS	ACS and subcomponents go offline-pending, then offline. LSMs must go offline before the ACS goes offline.
LSM	LSM goes offline-pending, then offline.
CAP	CAP goes offline-pending, then offline.
transport	If available, transport goes offline immediately. If in use, transport stays online.
port	Port goes offline immediately if: <ul style="list-style-type: none"> <li>• ACS has other ports online, or</li> <li>• ACS is offline.</li> </ul> Otherwise, the port stays online.

**vary offline force**

Table 31 shows the results of `vary offline force` for each ACS component. This option is useful if the device must be taken to the offline state while it is involved in extended activity (for example, during an audit).

**Table 31. Vary offline force Results**

ACS Component	Results
ACS	Only the current robotic request completes, then the ACS and subcomponents go offline immediately. Pending requests are discarded and new requests are rejected. LSMs must go offline before the ACS goes offline.
LSM	Only the current robotic request completes, then the LSM goes offline immediately. Pending requests are discarded and new requests are rejected.
CAP	Only the current robotic request completes, then the CAP goes offline immediately. Pending requests are discarded and new requests are rejected.
transport	Only the current robotic request completes, then the transport goes offline immediately. Pending requests are discarded and new requests are rejected.
port	Not valid.

**vary diagnostic**

The `vary diagnostic` request places the specified component in the diagnostic state (online to `cmd_proc` only). ACSLS processes all active and pending requests for the component, rejects new client application requests, then varies the component to diagnostic state. For an ACS, all of its subordinate LSMs are also varied to diagnostic.

**vary online**

Table 32 shows the results of `vary online` for each ACS component. When you vary a component online and it first goes to recovery state, ACSLS processes all active and pending requests for the component, rejects new requests, then varies the component online. When the component goes to online, ACSLS processes all requests for the component.

**Table 32. Vary Online Results**

ACS Component	Results
ACS	If ACS is offline, ACS and its LSM go to recovery, then online. If ACS is in diagnostic state, ACS and its LSM go to online immediately.

**Table 32. Vary Online Results**

<b>ACS Component</b>	<b>Results</b>
LSM	LSM goes to recovery, then online. You cannot vary an LSM online if it is attached to an offline ACS.
CAP	CAP goes to recovery, then online.
transport	Transport goes to recovery, then online.
port	Port goes online immediately.

**Examples**

To vary transport 0,0,9,3 offline:

```
vary drive 0,0,9,3 offline
```

To vary CAP 0,0,0 diagnostic:

```
vary cap 0,0,0 diagnostic
```

To force lsm 0,1 offline:

```
vary lsm 0,1 offline force
```

**Notes**

IPLing the system does not change the state of these components. Installing or reconfiguring ACSLS places all components in the online state wherever possible.

**See Also**

<b>For information about...</b>	<b>See...</b>
Displaying the status of a library component	<a href="#">“query commands” on page 337</a>

**Command Area Messages****Success Messages**

Vary: *type identifier* Varied *state*

**Explanation:** ACSLS changed the state of the specified library component.

**Variable:**

- *type* is the library component type.
- *identifier* is the library component identifier.
- *state* is the new stat of the library component.

### Intermediate Messages

None.

### Error Messages

In-transit cartridge recovery incomplete.

**Explanation:** The LSM robot cannot dispose of in-transit cartridges while the LSM or its ACS is varied online. If you are varying an ACS online, the ACS state is changed to online immediately, but any LSMs that are unable to complete in-transit cartridge recovery remain offline.

Vary: Vary type identifier failed, Drive in use.

**Explanation:** ACSLS cannot change the state of the specified library component because the command involved an in-use transport.

**Variable:**

- *type* is the library component type.
- *identifier* is the library component identifier.

Vary: Vary type identifier failed, Incorrect lockid

**Explanation:** ACSLS cannot change the state of the specified library component because the transport lock\_id does not match your lock\_id.

**Variable:**

- *type* is the library component type.
- *identifier* is the library component identifier.

Vary: Vary type identifier failed, State unchanged.

**Explanation:** ACSLS cannot change the state of the specified library component because the component is already in the requested state.

**Variable:**

- *type* is the library component type.
- *identifier* is the library component identifier.

Vary: Vary type identifier failed, Vary disallowed.

**Explanation:** ACSLS cannot change the state of the specified library component because a vary diagnostic request was issued from a client application.

**Variable:**

- *type* is the library component type.
- *identifier* is the library component identifier.

Vary: Vary type identifier failed, Incorrect state.

**Explanation:** ACSLS cannot change the state of the specified library component because you tried to vary:

- A CAP online when the LSM or ACS is in the diagnostic or offline state.
- A CAP diagnostic when the LSM or ACS is offline.
- An LSM online when the ACS is in the diagnostic or offline state.
- An LSM diagnostic if the ACS is offline.

**Variable:**

- *type* is the library component type.
- *identifier* is the library component identifier.

Vary: Vary type identifier failed, Vary in progress.

**Explanation:** ACSLS cannot change the state of the specified library component because

- A CAP and the CAP or its LSM or ACS is in a transitory (recovery or offline-pending) state.
- An LSM and the LSM, its ACS, or a subordinate CAP is in a transitory state.
- An ACS and the ACS, a subordinate LSM, or a subordinate CAP is in a transitory state.
- An ACS, LSM, or CAP to the diagnostic or online state and a vary offline/force request overrode the attempt.
- A transport which is in a transition state of recovery.

**Variable:**

- *type* is the library component type.
- *identifier* is the library component identifier.

Vary: Vary type identifier failed, State unchanged.

**Explanation:** ACSLS cannot change the state of the specified library component because you tried to vary an ACS, LSM, or CAP offline and a vary offline/force request overrode the attempt.

**Variable:**

- *type* is the library component type.
- *identifier* is the library component identifier.

Vary: Vary type identifier failed, ACS online.

**Explanation:** ACSLS cannot change the state of the specified library component because you tried to vary the last port offline in an online ACS.

**Variable:**

- `type` is the library component type.
- `identifier` is the library component identifier.

Vary: Vary type identifier failed, No ports online.

**Explanation:** ACSLS cannot change the state of the specified library component because there are no ports online for an ACS.

**Variable:**

- `type` is the library component type.
- `identifier` is the library component identifier.

**Display Area Messages**

One of the following messages appears when an ACS, LSM, transport, or port changes state.

```
ACS acs_id: state
LSM lsm_id: state
Drive drive_id: state
Port port_id: state
```

The following message appears if, during recovery of in-transit cartridges, a cartridge label is unreadable or duplicate. The cartridge is moved to the CAP.

```
CAP cap_id. Remove cartridges from CAP.
```

```
cap_id
  is the CAP containing the cartridges.
```

■ **venter**

The `venter` command makes a CAP ready to enter unlabeled cartridges into the library.

**Note:** The L5500, SL500, and SL8500 libraries do not support unlabeled cartridges (venters).

**Format**

```
venter cap_id vol_id
```

**Options**

```
cap_id
  specifies the CAP through which the cartridges are to be entered.
```

*vol\_id*

specifies the virtual labels of the cartridges to be entered.

### **Usage**

Use the `venter` command to make a CAP ready to enter cartridges with missing or unreadable labels into the library. You specify virtual labels (cartridge IDs) on the `venter` command, which adds cartridge information to the ACSLS database. From one to 42 cartridges can be entered at a time.

Unless the media type is “3480”, you cannot `venter` a cartridge that has no separate media type label.

**CAUTION:** ACSLS only supports the `venter` command for SCSI-attached LSMs when both:

- the cartridge being ventered has a media label, and
- the cartridge's media label is compatible with drives attached to the SCSI library.

Unless the cartridge's media type is 3480, you cannot `venter` a cartridge without a separate media label.

Use the `venter` command to enter cartridges with missing or unreadable labels. Do *not* open the LSM door and place cartridges with missing or unreadable labels in a storage cell because ACSLS cannot manage these cartridges. During an audit, ACSLS will eject cartridges with missing or unreadable labels and no virtual labels.

The `venter` command does not provide an option to specify the media type of the cartridge you want to enter. In a mixed-media environment, ACSLS cannot prevent transport/media incompatibilities for virtually entered cartridges!

### **Examples**

To assign virtual labels to cartridges MAINT1 and MAINT2 and enter these cartridges through CAP 0,2,2:

```
venter 0,2,2 MAINT1 MAINT2
```

**See Also**

<b>For information about...</b>	<b>See...</b>
Cancelling a command	<a href="#">“cancel” on page 300</a>
Creating or modifying scratch pools	<a href="#">“define pool” on page 306</a>
Ejecting cartridges from the library	<a href="#">“eject” on page 314</a>
Making a CAP (manual mode) ready to enter labelled cartridges into the library	<a href="#">“enter” on page 318</a>
Mounting a scratch cartridge on a transport	<a href="#">“mount *” on page 330</a>
Mounting a data cartridge on a transport	<a href="#">“mount” on page 327</a>
Displaying CAP status	<a href="#">“query cap” on page 340</a>
Setting CAP mode (manual or automatic)	<a href="#">“set cap mode” on page 365</a>
Setting CAP selection priority	<a href="#">“set cap priority” on page 367</a>

**Command Area Messages****Success Messages**

Venter: Enter complete, *nn* cartridges entered.

**Explanation:** ACSLS entered the displayed number of cartridges.

**Variable:** *nn* is the total number of cartridges entered.

Venter: *vol\_id* Entered through *cap\_id*

**Explanation:** ACSLS entered the specified virtual cartridge.

**Variable:**

- *vol\_id* is the virtual cartridge label.
- *cap\_id* is the CAP through which the cartridge was entered.

**Intermediate Messages**

None.

**Error Messages**

Venter: *vol\_id* Enter failed, *status*

**Explanation:** ACSLS did not enter the specified virtual cartridge.

**Variable:**

- *vol\_id* is the virtual cartridge label.
- The *status* is the translated completion status of the request returned by ACSLS.

**Display Area Messages**

CAP *cap\_id*: Place cartridges in CAP.

**Explanation:** The CAP is ready for cartridge entry. Open the CAP and enter cartridges.

**Variable:** *cap\_id* is the CAP used to enter cartridges.

CAP *cap\_id*: Unknown media type label.

**Explanation:** One or more cartridges cannot be entered because the media type is unknown.

**Variable:** *cap\_id* is the CAP used to enter cartridges. Open the CAP and remove the cartridges.



In this chapter you will learn how to use the display command.

The display command is a powerful tool for reporting information from the ACSLS database. Like a SQL SELECT statement, display allows you to specify:

- the database table (ACSLs object) you want to display
- wildcards
- conditions that objects must meet to be selected
- fields to report and the order in which they will be reported
- the sort order in which output will be displayed
- that only the count of the number of objects meeting the selection criteria be displayed

The result is a customizable query facility that lets customers specify what they want display and how it should be presented. Much more information is available through the display command than is possible through ACSLS query commands. For example, `display volume` gives access to volume information that was only available through a custom volrpt. `display lsm` and `display drive` provide LSM and tape drive serial numbers.

The display commands are:

[“display cap” on page 396](#)

Displays specific CAP information based on the options selected.

[“display cell” on page 398](#)

Displays specific cell information based on the options selected.

[“display drive” on page 399](#)

Displays specific drive information based on the options selected.

[“display lock” on page 401](#)

Displays specific lock information based on user ID.

[“display lsm” on page 403](#)

Displays specific LSM information based on the options selected.

[“display panel” on page 405](#)

Displays specific library panel information based on the options selected.

[“display pool” on page 406](#)

Displays specific scratch pool information based on the options selected.

[“display port” on page 408](#)

Displays specific port information based on the options selected.

[“display volume” on page 409](#)

Displays specific volume information based on the options selected.

## ■ display Commands

The `display` command displays various fields of information for different library components. The `display` command allows you to select the order and format in which the output is displayed.

Wildcards, numeric or alpha ranges, and selection lists can be used with the `display` command.

### Using Wildcard Characters

The wildcard character, `*`, can be used to substitute: for certain display operands; for one or more subfields in compound display operands; or at the beginning or end of a volume ID. It must match one or more characters.

Use `'_'` to match a single character in a *vol\_id*. The `'_'` is positional, and represents a single character. It may be used more than once in a *vol\_id* to represent more than one character.

- Examples of valid wildcards using the `*` character:

```
display cap * (displays all CAPs)
```

```
display volume *100 (displays all volumes ending in 100)
```

```
display drive * -volume RB0001 (displays the drive(s) that contain volume RB0001)
```

It is invalid to use the wildcard character, `*`: as part of a range; as one of multiple type operands; or embedded within a *vol\_id*. Use `'_'` to specify a single embedded character in a *vol\_id*.

- Examples of invalid wildcards using the `*` character:

```
display cap * * (Only one * is allowed when selecting all objects)
```

```
display volume 1*111 (The embedded * is not allowed. Use display volume 1_111 instead).
```

## Using Ranges

A range is denoted as “-”. Rules for using ranges are:

- Negative numbers are not allowed, except for volume home location.
- <n and >n are allowed. <n is all numerics less than n. >n is all numerics greater than n.
- Multiple numbers or ranges can be used, separated by a white space.
- Volume ranges can be alphanumeric.

Examples of valid ranges:

```
display volume * -drive 0,1,2,2-5 (displays all volumes in
acs 0, lsm 1, panel 2, drives 2 through 5)
```

```
display lsm * -state online -free_cells >50 -type 9730
(displays all 9730 lsms in an online state with a freecell count greater than
50)
```

Examples of invalid ranges:

```
display drive 0,1,1,1-* (* is not valid in a range)
```

```
display volume * -drive 0,1,?,1-5 (? is not valid in a range)
```

## Format

```
display type arg ... [ -selection ... ] [ [ -c ] | [ -f
field ... ] [ -s sort_field ... ] [ -n n ] ]
```

## Options

*type*

specifies the object to be displayed. Valid types are: cap, cell, drive, lock, LSMs, panel, pool, port, and volume.

*arg*

(argument) specifies the identifiers for the object type. Only objects that match one or more of the identifiers are returned.

*-selection*

specifies additional selection criteria. To be selected, a component must match at least one of the selection identifiers for each selection criteria.

*-c*

(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the *-f field*, *-s sort\_field*, and *-n n* options.

`-f field`

specifies the fields of information to be returned and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If `-f` is specified, the default fields are not displayed unless they are key fields or have been specified. If `-f` is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the `-f` option.

The list of fields is separated by spaces.

`-s sort_field`

sorts the objects returned by the field(s) selected in the order specified. The default sort order for each display command is discussed under each command. Sorting is based on internal database values and may not always display in alphanumeric order.

All returned fields for a type are valid `sort_fields`.

`-n n`

specifies the maximum number of objects to be displayed.

## ■ display cap

The `display cap` command displays specific CAP information based on the options selected.

### Format

```
display cap cap_id ... [ -status cap_status ... ]
[-priority cap_priority ... ] [ -state cap_state ... ]
[ -manual | -automatic ] [ -condition cap_condition ... ]
[ [ -c ] | [ -f field ... ] [ -s sort_field ... ] [ -n n ] ]
```

### Fields

The fields for `display cap` are:

- key fields: `acs`, `lsm`, `cap`
- default fields: `acs`, `lsm`, `cap`, `status`, `priority`, `state`, `mode`, `size`
- returned fields: `acs`, `lsm`, `cap`, `status`, `priority`, `state`, `mode`, `size`, `condition`

### Options

`cap_id`

specifies the CAP(s) to display. A `cap_id` is of the format: `acs`, `lsm`, `cap`.

Wildcard character, \*, can be used in any `cap_id` subfield or used to represent all subfields. However, if it is used to represent all `cap_ids` (example: `display cap *` or `display cap *,*,*`), additional `cap_ids` (either numeric or \*) are not allowed. Numeric ranges apply to any subfield of `cap_id`.

- status *cap\_status*  
specifies one or more CAP statuses as additional selection criteria. CAP statuses are: audit, available, eject, enter, insert\_magazines.
- priority *cap\_priority*  
specifies one or more CAP priorities. Numeric range rules apply. CAP numeric ranges: 0-16.
- state *cap\_state*  
specifies one or more CAP states. CAP states are: diagnostic, offline, offline\_pending, online, recovery.
- manual  
selects CAPs in manual mode.
- automatic  
selects CAPs in automatic mode.
- condition *cap\_condition*  
specifies one or more CAP conditions from which to select. Valid conditions are: inoperative, maint\_required, or operative.
- c  
(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the `-f` field, `-s` sort\_field, and `-n` *n* options.
- f *field*  
specifies the fields of information to be returned and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If `-f` is specified, the default fields are not displayed unless they are key fields or have been specified. If `-f` is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the `-f` option.  
  
The list of fields is separated by spaces.
- s *sort\_field*  
sorts the objects returned by the field(s) selected in the order specified. Sorting is based on internal database values and may not always display in alphanumeric order.  
  
All returned fields for a type are valid sort\_fields.
- n *n*  
specifies the maximum number of objects to be displayed.

## Examples

To display CAP data for ACS 1, LSM 1, CAP 1:

```
display cap 1,1,1
```

To display all manual mode CAPs in LSM 1, 0:

```
display cap 1,0,* -manual
```

To display all CAPs in an offline pending state:

```
display cap * -state offline_pending
```

## ■ display cell

The `display cell` command displays specific information about a cell based on the options selected.

## Format

```
display cell-id cell_loc ... [ -status cell_status ... ]
[ [ -c ] | [ -f field ... ] [ -s sort_field ... ]
[ -n n ] ]
```

## Fields

The fields for `display cell` are:

- key fields: acs, lsm, panel, row, column,
- default fields: acs, lsm, panel, row, column, status
- available fields: acs, lsm, panel, row, column, status

## Options

`-status cell_status`

specifies one or more cell statuses as additional selection criteria. Valid cell statuses: empty, full, inaccessible, reserved.

Wildcard character, \*, can be used in any `cell_id` subfield or used to represent all subfields. However, if it is used to represent all `cell_ids` (example: `display cell *` or `display cell_id *,*,*`), additional `cell_ids` (either numeric or \*) are not allowed. Numeric ranges apply to any subfield of `cell_id`.

`-c`

(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the `-f field`, `-s sort_field`, and `-n n` options.

`-f field`

specifies the fields of information and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If `-f` is specified, the default fields are not displayed unless they are key fields or have been specified. If `-f` is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the `-f` option.

The list of fields is separated by spaces.

`-s sort_field`

sorts the objects returned by the field(s) selected in the order specified. Sorting is based on internal database values and may not always display in alphanumeric order.

All returned fields for a type are valid `sort_fields`.

`-n n`

specifies the maximum number of objects to be displayed.

## Example

To display information for cells with a reserved status:

```
display cell *,*,*,*,* -status reserved
```

## ■ display drive

The `display drive` command displays specific drive information based on the options selected.

## Format

```
display drive drive_id ... [ -status drive_status ... ]
[-state drive_state ... ] [ -type drive_type ... ]
[ -volume vol_id ... ] [ -lock lock_id... ] [ -serial
drive_serial_num ... ] [ -condition drive_condition ...]
[ [ -c ] | [ -f field ... ] [ -s sort_field ... ] [ -n n ] ]
```

## Fields

The fields for `display drive` are:

- key fields: acs, lsm, panel, drive
- default fields: acs, lsm, panel, drive, status, state, volume, type
- available fields: acs, lsm, panel, drive, status, state, volume, type, lock, serial\_num, condition

ACSLs can only report drive serial numbers when the drive reports its serial number to the library and the library reports its serial number to ACSLS.

Only modern libraries and drives report the drive serial numbers to ACSLS. This includes T9840, T9940, LTO, DLT 7000, SDLT, and later drives.

SCSI-attached libraries, L5500, and later libraries report serial numbers to ACSLS.

## Options

*drive\_id*

displays the *drive\_id* in the following format: *acs, lsm, panel, drive*.

Wildcard character, \*, can be used in any *drive\_id* subfield or used to represent all subfields. However, if it is used to represent all *drive\_ids* (example: *display drive \** or *display drive\_id \*,\*,\*,\**), additional *drive\_ids* (either numeric or \*) are not allowed. Numeric ranges apply to any subfield of *drive\_id*.

-status *drive\_status*

displays one of the following statuses as additional selection criteria: available, in\_use, reserved

**Possible scenario:** You perform a *query drive all* and get a message saying the drive is in use. You then perform a *display drive \** and get a message saying the drive is reserved. This means: the reserved status for the drive implies that a mount request is in progress and that a volume is on its way to the drive. At the same time the drive is considered to be in use.

-state *drive\_state*

displays the following states for one or more drives: diagnostic, online, offline, recovery.

-type *drive\_type*

displays drives by transport type.

-volume *vol\_id*

displays volume(s) mounted on the drive(s) selected. A *vol\_id* can be a 6-alphanumeric string or wildcard string that specifies a tape volume volume or range of volume IDs.

-lock *lock\_id*

displays drive information on locked drives by lock ID.

-serial *drive\_serial\_num*

specifies the drive serial number.

-condition *drive\_condition#*

displays condition of the drive(s) specified. Valid conditions are: operative, inoperative, or maint\_required.

- `-c`  
(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the `-f field`, `-s sort_field`, and `-n n` options.
- `-f field`  
specifies the fields of information and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If `-f` is specified, the default fields are not displayed unless they are key fields or have been specified. If `-f` is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the `-f` option.
- The list of fields is separated by spaces.
- `-s sort_field`  
sorts the objects returned by the field(s) selected in the order specified. Sorting is based on internal database values and may not always display in alphanumeric order.
- All returned fields for a type are valid `sort_fields`.
- `-n n`  
specifies the maximum number of objects to be displayed.

## Examples

To display a maximum of 56 rows for all drives in ACS 1, sorted by panel and drive:

```
display drive 1,*,*,* -s panel drive -n 56
```

To display drive data for ACS 1, LSM 1, with status available:

```
display drive 1,1,*,* -status available
```

To display drive types and serial numbers for all drives:

```
display drive *,*,*,* -f type serial_num
```

## ■ display lock

The `display lock` command displays specific lock information by *lock\_id* based on the options selected.

## Format

```
display lock lock_id ... [ -user user_id ... ] [ [ -c ] |  
[ -f field ... ] [ -s sort_field ... ] [ -n n ] ]
```

## Fields

The fields for display lock are:

- key fields: lock\_id
- default fields: lock\_id, user\_id
- available fields: lock\_id, user\_id

## Options

*lock\_id*

displays the numeric lock\_id.

Wildcard character,\*, can be used in any lock\_id subfield or used to represent all subfields. However, if it is used to represent all lock\_ids (example: *display lock \** or *display lock\_id \*,\*,\**), additional lock\_ids (either numeric or \*) are not allowed. Numeric ranges apply to any subfield of lock\_id.

*user\_id*

specifies the user name (example: acsss) that has locked the volume or transport. The *user\_id* wraps after 80 characters.

-c

(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the -f field, -s sort\_field, and -n n options.

-f *field*

specifies the fields of information and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If -f is specified, the default fields are not displayed unless they are key fields or have been specified. If -f is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the -f option.

The list of fields is separated by spaces.

-s *sort\_field*

sorts the objects returned by the field(s) selected in the order specified. Sorting is based on internal database values and may not always display in alphanumeric order.

All returned fields for a type are valid sort\_fields.

-n *n*

specifies the maximum number of objects to be displayed.

## Example

To display lock information for lock\_id 2:

```
display lock 2
```

## ■ display lsm

The `display lsm` command displays specific LSM information based on the options selected.

## Format

```
display lsm lsm_id ... [ -status lsm_status ... ]
[-state lsm_state ... ] [ -free_cells cell_count ... ]
[ -type lsm_type ... ] [ -serial lsm_serial_num ... ]
[ -condition lsm_condition ] [ -door_open | -door_closed ]
[ [ -c ] | [ -f field ... ] [ -s sort_field ... ]
[ -n n ] ]
```

## Fields

The fields for `display lsm` are:

- key fields: `acs`, `lsm`
- default fields: `acs`, `lsm`, `status`, `state`, `free_cells`
- available fields: `acs`, `lsm`, `status`, `state`, `free_cells`, `type`, `serial_num`, `condition`, `door_status`

*lsm\_id*

displays the lsm id in the following format: *acs*, *lsm*.

Wildcard character, `*`, can be used in any `lsm_id` subfield or used to represent all subfields. However, if it is used to represent all `lsm_ids` (example: `display lsm *` or `display lsm_id *,*`), additional `lsm_ids` (either numeric or `*`) are not allowed. Numeric ranges apply to any subfield of `lsm_id`.

`-status lsm_status`

specifies one or more LSM statuses from which to select. Valid statuses are: `audit`, `normal`

`-state lsm_state`

specifies one or more LSM states from which to select. Valid states are: `diagnostic`, `online`, `offline`, `offline_pending`, `recovery`.

`-free_cells cell_count`

selects LSMs by the number of free cells.

- type *lsm\_type*  
displays LSMs by type.
- serial *lsm\_serial\_num*  
displays serial numbers for selected LSMs.
- condition *lsm\_condition*  
displays condition of the selected LSMs. Valid conditions are: operative, inoperative, or maint\_required.
- door\_open  
displays LSMs with open doors.
- door\_closed  
displays LSMs with closed doors.
- c  
(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the -f field, -s sort\_field, and -n n options.
- f *field*  
specifies the fields of information and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If -f is specified, the default fields are not displayed unless they are key fields or have been specified. If -f is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the -f option.  
  
The list of fields is separated by spaces.
- s *sort\_field*  
sorts the objects returned by the field(s) selected in the order specified. Sorting is based on internal database values and may not always display in alphanumeric order.  
  
All returned fields for a type are valid sort\_fields.
- n *n*  
specifies the maximum number of objects to be displayed.

## Example

To display information for all LSMs in the library for type 9714:

```
display lsm * -type 9714
```

## ■ display panel

The `display panel` command displays specific library panel information based on the options selected.

### Format

```
display panel panel_id ... [ -type panel_type ... ] [ [ -c ] |
[ -f field ... ] [ -s sort_field ... ] [ -n n ] ]
```

### Fields

The fields for display panel are:

- key fields: `acs`, `lsm`, `panel`
- default fields: `acs`, `lsm`, `panel`, `type`
- available fields: `acs`, `lsm`, `panel`, `type`

### Options

*panel\_id*

displays the panel id in the following format: *acs*, *lsm*, *panel*.

Wildcard character, `*`, can be used in any `panel_id` subfield or used to represent all subfields. However, if it is used to represent all `panel_ids` (example: `display panel *` or `display panel_id *,*,*`), additional `panel_ids` (either numeric or `*`) are not allowed. Numeric ranges apply to any subfield of `panel_id`.

`-type panel_type`

specifies one or more valid panel types.

`-c`

(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the `-f field`, `-s sort_field`, and `-n n` options.

`-f field`

specifies the fields of information and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If `-f` is specified, the default fields are not displayed unless they are key fields or have been specified. If `-f` is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the `-f` option.

The list of fields is separated by spaces.

`-s sort_field`  
 sorts the objects returned by the field(s) selected in the order specified. Sorting is based on internal database values and may not always display in numeric order.

All returned fields for a type are valid `sort_fields`.

`-n n`  
 specifies the maximum number of objects to be displayed.

## Example

Display panel data for all panels:

```
display panel *
```

## ■ display pool

The `display pool` command displays specific scratch pool information based on the options selected.

## Format

```
display pool pool_id ... [ -low_water low_water_mark ... |
  -high_water high_water_mark... ]
  [-overflow | -no_overflow ] [ [ -c ] | [ -f field ... ]
  [ -s sort_field ... ] [ -n n ] ]
```

## Fields

The fields for `display pool` are:

- key fields: `pool_id`
- default fields: `pool_id`, `low_water`, `high_water`, `overflow`
- available fields: `pool_id`, `low_water`, `high_water`, `overflow`

## Options

`pool_id`  
 displays the pool id of a scratch pool.

Wildcard character, `*`, can be used in any `pool_id` subfield or used to represent all subfields. However, if it is used to represent all `pool_ids` (example: `display pool *` or `display pool_id *,*,*`), additional `pool_ids` (either numeric or `*`) are not allowed. Numeric ranges apply to any subfield of `pool_id`.

- low\_water *low\_water\_mark*  
specifies the minimum number of scratch cartridges in the pool. If a low\_water mark is displayed with a hyphen (-) appended to it, the number of cartridges in the library for the selected scratch pool is within the designated low\_water mark.
- high\_water *high\_water\_mark*  
specifies the maximum number of scratch cartridges in the pool. If a high\_water mark is displayed with a hyphen (-) appended to it, the number of cartridges in the library for the selected scratch pool is within the designated high\_water mark.
- overflow  
selects pools wherein if a scratch mount is requested and if the pool is empty, the common pool is used to satisfy the request.
- no\_overflow  
selects pools wherein if a scratch mount is requested and if the pool is empty, the mount fails without checking the common pool for available tapes.
- c  
(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the -f field, -s sort\_field, and -n n options.
- f *field*  
specifies the fields of information and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If -f is specified, the default fields are not displayed unless they are key fields or have been specified. If -f is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the -f option.  
  
The list of fields is separated by spaces.
- s *sort\_field*  
sorts the objects returned by the field(s) selected in the order specified. Sorting is based on internal database values and may not always display in numeric order.  
  
All returned fields for a type are valid sort\_fields.
- n *n*  
specifies the maximum number of objects to be displayed.

## Example

To display all information for pool 5:

```
display pool 5
```

To display all pools with overflow:

```
display pool * -overflow
```

## ■ display port

The `display port` command displays specific port information based on the options selected.

## Format

```
display port port_id ... [ -online | -offline ]
[ -name port_name ... ] [ [ -c ] | [ -f field ... ]
[-s sort_field ... ] [ -n n ] ]
```

## Fields

The fields for display port are:

- key fields: acs, port
- default fields: acs, port, name, state
- available fields: acs, port, name, state

## Options

*port\_id*

displays the port id in the following format: *acs, port*.

Wildcard character, \*, can be used in any *port\_id* subfield or used to represent all subfields. However, if it is used to represent all *port\_ids* (example: *display port \** or *display port\_id \*,\**), additional *port\_ids* (either numeric or \*) are not allowed. Numeric ranges apply to any subfield of *port\_id*.

- online  
selects and displays ports in an online state.
- offline  
selects and displays ports in an offline state.
- name *port\_name*  
specifies the port name.

- c  
(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the -f field, -s sort\_field, and -n n options.
- f *field*  
specifies the fields of information and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If -f is specified, the default fields are not displayed unless they are key fields or have been specified. If -f is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the -f option.
- The list of fields is separated by spaces.
- s *sort\_field*  
sorts the objects returned by the field(s) selected in the order specified. Sorting is based on internal database values and may not always display in alphanumeric order
- All returned fields for a type are valid sort\_fields.
- n *n*  
specifies the maximum number of objects to be displayed.

## Example

To display all ports in an online state:

```
display port * -online
```

## ■ display volume

The `display volume` command displays specific volume information based on the options selected.

## Format

```
display volume vol_id ... [ -home acs,lsm,panel,row,column... ]
[ -drive drive_loc ... ] [ -data | -scratch | -clean ]
[ -media media_type ... ] [ -pool pool_id... ] [ -standard |
-virtual ] [ -status vol_status ... ] [ -entry entry_date ... ]
[ -access access_date ... ] [ -lock lock_id ... ] [ [ -c ] |
[ -f field ... ] [ -s sort_field ... ] [ -n n ] ]
[ -max_use max_use ] [ -lock_time lock_time ]
```

## Fields

The fields for `display volume` are:

- key field: `vol_id`
- default fields: `vol_id`, `acs`, `lsm`, `panel`, `row`, `column`, `pool`, `status`, `media`, `type`
- available fields: `vol_id`, `acs`, `lsm`, `panel`, `row`, `column`, `drive`, `type`, `media`, `pool`, `label_type`, `status`, `entry_date`, `access_date`, `access_count`, `max_use`, `lock`, `lock_time`

## Options

`vol_id`

displays the volume(s) selected.

A `vol_id` can be one or more 1-6 alphanumeric strings that specifies a tape cartridge.

The wildcard character, `*`, can be used to represent all `vol_ids` (example: `display vol *`) or to represent a group of cartridges (example: `display vol_id VAP*` displays all cartridges starting with VAP).

The `vol_id` may contain any valid volume ID characters and `*`.

`-home home_loc`

specifies volume home location(s) from which to retrieve volume information in the following format: `acs,lsm,panel,row,column`. It cannot be used with `-drive`.

`home_loc` subfields can use numeric ranges. All `home_loc` subfields must be specified. A wildcard character, `*`, can be used in place of one or more `drive_loc` subfields. Numeric range rules apply to any subfield or `drive_loc`.

`-drive drive_loc`

specifies drive location(s) from which to retrieve volume information in the following format: `acs,lsm,panel,drive`. It cannot be used with `-home`

`drive_loc` subfields can use numeric ranges. All `home_loc` subfields must be specified. A wildcard character, `*`, can be used in place of one or more `drive_loc` subfields. Numeric range rules apply to any subfield or `drive_loc`.

`-data`

selects data cartridges.

`-scratch`

selects cartridges in scratch mode.

`-clean`

selects cleaning cartridges.

- media *media\_type*  
selects only cartridges with the specified media type.
- pool *pool\_id*  
selects and displays cartridges from scratch pools. Cartridges retain the ID of the scratch pool even after they have changed to data cartridges.
- standard  
selects a tape(s) with a scannable volume ID *label\_type*. It cannot be used with *-virtual*.
- virtual  
selects a tape(s) without an external volume ID *label\_type* that was entered through the *venter* command. It cannot be used with *-standard*.
- status *vol\_status*  
valid entries are: *dismount, eject, enter, mount, home, in\_drive, move, missing, absent, ejected*.
- entry *entry\_date*  
specifies the date and time the cartridge was entered into the library. It is displayed in ISO standard format.
- access *access\_date*  
specifies the date and time the cartridge was last used. It is displayed in ISO standard format.
- lock *lock\_id*  
specifies the lock ID for the volume.
- max\_use *max\_use*  
specifies the maximum number of uses for a cleaning cartridge. Non-cleaning cartridges have a *max\_use* value of zero.
- lock\_time *lock\_time*  
specifies the date and time the lock was set. It is displayed according to the system default format.
- c  
(count) causes the request to display only the number of objects that satisfy the *arg* and *selection* criteria. This option is mutually exclusive with the *-f field*, *-s sort\_field*, and *-n n* options.
- f *field*  
specifies the fields of information and the order in which they are to be returned for each object selected. Each display type has a list of available, key, and default fields. If *-f* is specified, the default fields are not displayed unless they are key fields or have been specified. If *-f* is not specified, then the default fields are returned. Key fields for each type are normally displayed first. However, you can change the order in which the key fields are displayed by specifying them after the *-f* option.

The list of fields is separated by spaces.

- s *sort\_field*  
sorts the objects returned by the field(s) selected in the order specified. Sorting is based on internal database values and may not always display in alphanumeric order  
  
All returned fields for a type are valid sort\_fields.
- n *n*  
specifies the maximum number of objects to be displayed.

## Examples

To display all volumes entered this year:

```
display volume * -entry 2002
```

To display all volumes entered in January, 2002

```
display volume * -entry 2002-01
```

To display all volumes entered on January 27, 2002

```
display volume * -entry 2002-01-27
```

To display all volumes entered on January 27, 2002 between 10:00 a.m. and 11:00 a.m.

```
display volume * -entry 2002-01-27:10
```

To display all volumes entered between January 2002 and April 2002.

```
display volume * -entry 2002-01-2002-04
```

To display all volumes entered between 6:33 a.m. and 6:57:45 April 10 (for example, you may want to do this after an outage).

```
display volume * -entry 2002-04-10:06:33:00-2002-04-10:6:57:45
```

To display all volumes entered since February 25, 2002.

```
display volume * -entry >2002-02-25
```

To display all 9840 data volumes.

```
display volume * -media STK1R
```

To display a cleaning volumes by ACS and media type:

```
display volume * -home acs,*,*,*,* -media media_type  
-f vol_id acs lsm media max_use access_count
```

# Second Disk Support

## A

---

In this chapter, you will learn how to:

- Configure the second disk on Solaris
- De-install the second disk Solaris
- Configure the second disk on AIX
- De-install the second disk on AIX

## ■ Configuring the Second Disk on Solaris

**Note:** If you have already partitioned the second disk when you installed the operating system, go to [“Installing ACSLS Second Disk Support” on page 419](#).

If you did not partition the second disk when you installed the operating system, or you are installing a new second disk, complete this section.

## Space Required by ACSLS on the Second Disk

Second disk is used to contain the transaction logs in order to assure recoverability to the most recent transaction, even in the event of primary disk failure. The storage requirements for second disk can be met with 256 MB in Partition 0 and the remainder in Partition 1.

## Partitioning the Second Disk

In this procedure, you create two partitions.

**Note:** This procedure assumes a 4MB disk.

### To partition the second disk:

1. Is ACSLS running?

---

YES Continue with Step 2.

---

NO Go to Step 7.

---

2. Open a terminal or a command tool and log in as `acssa`.

3. From the `cmd_proc` window, idle ACSLS.

```
idle
```

4. Open another command tool and log in as `acsss`.

5. Shut down ACSLS.

```
kill.acsss
```

6. Shut down the database.

```
db_command stop
```

7. Log in as `root`.

8. Open a terminal window. At the prompt, enter

```
format
```

The system prompts you to select a disk. On a SPARC or Ultra system, the second disk is typically `c0t1d0`.

9. Enter the number of the second disk (typically menu number 1).

The Format Menu displays, shown in [Figure 45 on page 415](#).

**Figure 45. Format Menu**

```

FORMAT MENU:

disk - select a disk
type - select (define) a disk type
partition- select (define) a partition table
current- describe the current disk
format- format and analyze the disk
repair- repair a defective sector
label- write label to the disk
analyze- surface analysis
defect- defect list management
backup- search for backup labels
verify- read and display labels
save - save new disk/partition definitions
inquiry- show vendor, product and revision
volname- set 8-character volume name
quit

```

10. Enter partition from the Format Menu.

The Partition Menu displays, shown in [Figure 46](#).

**Figure 46. Partition Menu**

```

PARTITION MENU:

0 - change '0' partition
1 - change '1' partition
2 - change '2' partition
3 - change '3' partition
4 - change '4' partition
5 - change '5' partition
6 - change '6' partition
7 - change '7' partition
select- select a predefined table
modify- modify a predefined partition table
name - name the current table
print- display the current table
label- write partition map and label to the disk
quit

```

11. Enter print to display the disk partitions.

The Current Partition Table displays, shown in [Figure 47 on page 416](#).

**Figure 47. Partition Table**

```

Current partition table (original)

      Total disk cylinders available: 2036+2 (reserved cylinders)

Part TagFlag      CylindersSize      Blocks
0   unassignedwm  0-2035  1002.09MB  (2036/0/0)
1   unassignedwm  0        0          (0/0/0)
2   unassignedwm  0        0          (0/0/0)
3   unassignedwm  0        0          (0/0/0)
4   unassignedwm  0        0          (0/0/0)
5   unassignedwm  0        0          (0/0/0)
6   unassignedwm  0        0          (0/0/0)
7   unassignedwm  0        0          (0/0/0)

```

**Note:** The values in your partition table might differ from those shown, depending on the size of your disk and whether you have previously partitioned it.



**CAUTION:** Modifying an existing partition will destroy any data in that partition.

12. Enter 0 to change the 0 partition.

The partition 0 table is displayed. Enter the values shown.

a. Enter partition id tag [unassigned]:

Accept either the default unassigned or enter `he lp` to bring up a list of available id tags.

b. Enter partition permission flags [wm]: `wm`

c. Enter new starting cyl [0]: `0`

d. Enter partition size [0b, 0c, 0.00mb]: `400m`

13. Enter `print` to confirm changes made to partition 0.

14. Note the number of cylinders in partition 0.

Subtract the number of cylinders in partition 0 from the total number of disk cylinders. This number will be used in Step 15.

15. Enter 1 to change partition 1.

The partition 1 table is displayed. Enter the values shown.

a. Enter partition id tag [unassigned]:

Accept the default.

b. Enter partition permission flags [wm]: `wm`

c. Enter new starting cyl [0]:

To determine this value, add 1 to the highest numbered cylinder in partition 0. For example, if partition 0 contains cylinders 0-520, enter 521

d. Enter partition size [0b, 0c, 0.00mb]:

16. Enter `print` to confirm the changes entered.

The display should now look like the configuration shown in [Figure 48](#).

The remaining partitions should all be unassigned with starting cylinder of 0 and partition size of 0.

**Figure 48. Second Disk Partition Table**

```

Current partition table (original)

    Total disk cylinders available: 2036 + 2 (reserved cylinders)

PartTagFlag      CylindersSize      Blocks
0  unassignedwm    0-520    256MB      (520/0/0)
1  unassignedwm    521-2035 17xxMB      (1515/0/0)
2  unassignedwm    0         0           (0/0/0)
3  unassignedwm    0         0           (0/0/0)
4  unassignedwm    0         0           (0/0/0)
5  unassignedwm    0         0           (0/0/0)
6  unassignedwm    0         0           (0/0/0)
7  unassignedwm    0         0           (0/0/0)

```

**Note:** On your display, `unassigned` should appear on size 0 entries.

17. Enter `?` to display the partition menu.

The Partition Menu appears, shown in [Figure 49 on page 418](#).

**Figure 49. Partition Menu**

```

PARTITION MENU:

0   - change '0' partition
1   - change '1' partition
2   - change '2' partition
3   - change '3' partition
4   - change '4' partition
5   - change '5' partition
6   - change '6' partition
7   - change '7' partition
select- select a predefined table
modify- modify a predefined partition table
name - name the current table
print- display the current table
label- write partition map and label to the disk
quit

```

18. Enter `label` from the partition menu to update the disk with the new partitions.
19. Enter `y` at the Ready to label disk, continue? prompt.
20. From the partition prompt, enter `<CTRL>+D` to exit.

## Creating the File Systems

In this procedure, you create the file systems on the disk you just partitioned.

### To create file systems:

1. At the prompt, enter `y` to construct a new filesystem.
2. At the system prompt, enter

```
newfs /dev/dsk/devname1
```

where `devname1` is `cntndns0`, and `cntndn` matches the disk you just installed.

For example,

```
newfs /dev/dsk/c0t2d0s0
```

The system may display:

```
/dev/rdisk/c0t2d0s0 last mounted as /second_disk
```

3. To have second disk file systems automatically mounted on boot up, use an editor (such as `vi` or `textedit`) to edit the `/etc/vfstab` file to define the mount points. Note that each field is separated by tabs.

Add entries to the file such as the following two entries:

```
/dev/dsk/c0t1d0s0 /dev/rdisk/c0t1d0s0 /second_disk ufs 2 yes -
/dev/dsk/c0t1d0s1 /dev/rdisk/c0t1d0s1 /second_disk/backup ufs 2 yes -
```

4. Verify that there is a directory for your mount point `/second_disk` by entering:

```
ls -ld /second_disk
```

If the directory does not exist, create one by entering

```
mkdir /second_disk
```

5. Make sure that the second disk is mounted:

```
mount /second_disk
```

6. Make a directory for the backup portion of the second disk.

```
mkdir /second_disk/backup
```

This directory is the default directory for the backup portion of the second disk. You may create another directory name for second disk backups.

7. Make sure the second disk backup is mounted on the directory specified in the previous step.

```
mount /second_disk/backup
```

## Installing ACSLS Second Disk Support

To install second disk support:

1. Is ACSLS running?

---

YES Continue with Step 2.

---

NO Go to Step 6.

---

2. Open a command tool and log in as `acsss`.

3. From the `cmd_proc` window, idle ACSLS.

```
idle
```

4. Shut down ACSLS.

```
kill.acsss
```

5. Shut down the database.

```
db_command stop
```

6. Log in as root.
7. Change directories.

```
cd /export/home/ACSSS/install
```
8. Run the second disk utility.

```
./sd_mgr.sh
```
9. Choose 1 to install second disk:

```
1: Installation of a second database disk.
```
10. When you are prompted for the directory for second disk support, press <Enter> to accept the default of /second\_disk if you used /second\_disk as the mount point in your vfstab entry (see [“Creating the File Systems” on page 418](#)). If you used an optional directory name instead of /second\_disk, enter that name.
11. When you are prompted for the second disk directory for database backups, press <Enter> to accept the default of /second\_disk/backup if you used /second\_disk/backup as the mount point in your vfstab entry (see [“Creating the File Systems” on page 418](#)). If you used an optional directory name instead of /second\_disk/backup, enter that name.
12. When you see the Second Disk Support menu, select 3 to exit the system.

You should see a message about doing a mandatory backup and a prompt to insert a database backup tape.
13. Insert the tape if you need to and wait for the backup to complete.
14. When the prompt returns, log out.
15. Restart ACSLS.

## Restarting ACSLS

### To restart ACSLS:

1. Log in as root.
2. In a terminal window or at the shell prompt, enter

```
reboot
```

**Note:** If the automatic startup on reboot option was enabled during ACSLS installation, ACSLS is initiated automatically when the server system is rebooted. If this is the case, go to Step 3.

3. Log in as `acsss`.
4. In a terminal window or at the shell prompt, enter  

```
rc.acsss
```

It takes a few seconds to start the database and return to the shell prompt.
5. When the shell prompt returns, log out.
6. Log back in as `acssa` to use the ACSLS software.

## ■ De-installing a Second Disk on Solaris

De-installing a second disk is usually done when the second disk needs to be serviced or replaced.



**CAUTION:** De-installation of the second disk should be used if there is a hardware failure of the second disk. ACSLS will not start if a hardware failure occurs.

### To de-install the second disk:

1. If you are not already logged in as `acsss` log in as `acsss`.
2. From the `cmd_proc` window, idle ACSLS.  

```
idle
```
3. Shut down ACSLS.  

```
kill.acsss
```
4. Shut down the database.  

```
db_command stop
```
5. Log in as `root`.
6. Change directories by entering:  

```
cd /export/home/ACSSS/install
```
7. Run the second disk utility.  

```
./sd_mgr.sh
```

This command starts the second disk manager. With this function, you can remove second disk support.
8. Enter 2 to de-install second disk support when you are prompted for a menu choice.  

```
2: De-installation of a second database disk.
```
9. Respond `y` or `n` when you see the following prompt:

```
Are you de-installing a bad secondary disk?  
(y or n):
```

Select y if your second disk is damaged and non-functioning.

Select n if your second disk is not damaged.

10. When the Second Disk Support menu appears, select 3 to exit the menu.  
A message and prompt display indicating you must do a mandatory backup and to insert a database backup tape.
11. Insert the tape and wait for the backup to complete.  
The backup takes several minutes to complete.
12. When the backup is completed, remove the backup tape and store it in a safe place.
13. Log out by selecting Log Out from the Workspace menu.
14. Restart ACSLS.

## ■ Configuring the Second Disk on AIX

### Partitioning the Second Disk

In this procedure, you create two partitions on the second disk.

**To partition the second disk:**

1. Open a Command tool and log in as `acssa`.
2. From the `cmd_proc` window, idle ACSLS.  
`idle`
3. Open another command tool and log in as `acsss`.
4. Shut down ACSLS:  
`kill.acsss`
5. Shut down the database:  
`db_command stop`
6. Log in as `root`.
7. To determine the name of the new disk, enter  
`lsdev -C -s scsi`

The system displays a list of all SCSI devices, which can include the tape drive, the internal disk drive(s), and the disk drive added for second disk support.

Typically, the name of the second disk is `hdisk1`. Verify that it displays as Available.

8. At the console prompt, enter  
`smitty mkvg`  
 The Add a Volume Group screen appears.
9. Select VOLUME GROUP name and enter  
**second\_disk**
10. Select PHYSICAL VOLUME names and
  - a. Press <F4>.
  - b. Select `hdisk1`.
11. Press <Enter> to run the command.  
 The COMMAND STATUS screen appears, and the system runs the command.
12. When the Command: OK message appears at the top of the COMMAND STATUS screen, press <F10> to return to the prompt.

## Creating Second Disk File Systems

To create second disk file systems:

1. At the console prompt, enter  
`smitty crjfs`  
 The Add a Journalled File System screen appears.
2. Select the Add a Standard Journalled File System option.  
 The Volume Group Name screen appears identifying the volume group name.
3. Select `second_disk` and press <Enter>.
4. In the Size of file system (in 512-byte blocks) field, enter 921600.
5. In the MOUNT POINT field, enter `/second_disk`.
6. In the Mount AUTOMATICALLY at system restart? field, press <F4> select yes.
7. Press <Enter> to run the command.  
 The COMMAND STATUS screen appears, and the system runs the command.

8. When the Command: OK message appears at the top of the COMMAND STATUS screen, press <F10> to return to the prompt.

## Mounting Second Disk File Systems

### To mount second disk file systems:

1. At the console prompt, enter  
smitty mountfs.  
The Mount a File System screen appears.
2. In the FILE SYSTEM name field
  - a. Press <F4>.
  - b. Select the item on the same line as /second\_disk.
  - c. Press <Enter>.
3. In the DIRECTORY over which to mount field
  - a. Press <F4>.
  - b. Select /second\_disk.
  - c. Press <Enter>.
4. In the TYPE of file system field
  - a. Press <F4>.
  - b. Select jfs.
  - c. Press <Enter>.
5. Press <Enter> to run the command.  
The COMMAND STATUS screen appears, and the system runs the command.
6. When the Command: OK message appears at the top of the COMMAND STATUS screen, press <F10> to return to the prompt.

## Creating Second Disk Backup File Systems

### To create second disk backup file systems:

1. At the console prompt, enter  
smitty crjfs  
The Add a Journalled File System screen appears.
2. Select the Add a Standard Journalled File System option.

The `VOLUME GROUP NAME` screen appears identifying the volume group name.

3. Select `second_disk` and press `<Enter>`.
4. In the `Size of file system (in 512-byte blocks)` field, enter a value determined as follows:
  - If the library is a 9310 (5500 volumes) or smaller, enter 5112000.
  - Otherwise, multiply 400000 by the number of 9310s and enter the result. For example, if you have six 9310s, you multiply  $6 \times 400000 = 2400000$ .

**Note:** If the `/second_disk/backup` file system becomes full, use `smitty chjfs` to increase the size. Increase the size to account for your system's activity and your ACSLS backup retention period (if longer than eight days).
5. In the `MOUNT POINT` field, enter `/second_disk/backup`.
6. In the `Mount AUTOMATICALLY at system restart?` field;
  - a. Press `<F4>`.
  - b. Select `yes`.
  - c. Press `<Enter>`.
7. Press `<Enter>` to run the command.
 

The `COMMAND STATUS` screen appears, and the system runs the command.
8. When the `Command: OK` message appears at the top of the `COMMAND STATUS` screen, press `<F10>` to return to the prompt.

## Mounting Second Disk Backup File Systems

### To mount second disk backup file systems:

1. At the console prompt, enter
 

```
smitty mountfs
```

The `Mount a File System` screen appears.
2. In the `FILE SYSTEM name` field
  - a. Press `<F4>`.
  - b. Select the item on the same line as `/second_disk/backup`.
  - c. Press `<Enter>`.
3. In the `DIRECTORY over which to mount` field

- a. Press <F4>.
  - b. Select the item on the same line as /second\_disk/backup.
  - c. Press <Enter>.
4. In the TYPE of file system field
- a. Press <F4>.
  - b. Select jfs.
  - c. Press <Enter>.
5. Press <Enter> to run the command.
- The COMMAND STATUS screen appears, and the system runs the command.
6. When the Command: OK message appears at the top of the COMMAND STATUS screen, press <F10> to return to the prompt.

## Installing Second Disk Support

### To install the second disk:

1. If you are not already logged in as acsss, select Log Out from the Workspace menu and then log in as acsss.
2. From the cmd\_proc window, idle ACSLS.  
idle
3. Shut down ACSLS:  
kill.acsss
4. Shut down the database:  
db\_command stop
5. Log in as root.
6. Change directories by entering  
cd /export/home/ACSSS/install
7. Execute the second disk utility by entering  
./sd\_mgr.sh
8. Choose 1 to install second disk:  
1: Installation of a second database disk.
9. When you are prompted for the directory for second disk support, press <Enter> to accept the default of /second\_disk if you used

`/second_disk` as the mount point when you created the second disk file system (see [“Creating Second Disk File Systems” on page 423](#)).

10. When you are prompted for the second disk directory for database backups, press <Enter> to accept the default of `/second_disk/backup` if you used `/second_disk/backup` as the mount point when you created second disk backup file systems (see [“Creating Second Disk Backup File Systems” on page 424](#)).

11. When you see the Second Disk Support menu, select 3 to exit the system.

You should see a message about doing a mandatory backup and a prompt to insert a database backup tape.

12. Insert the tape if you need to and wait for the backup to complete.
13. When the prompt returns, log out.

## Restarting ACSLS

### To restart ACSLS:

1. Log in as root.
2. In a terminal window or at the shell prompt, enter  
`reboot`

**Note:** If the automatic startup on reboot option was enabled during ACSLS installation, ACSLS is initiated automatically when the server system is rebooted. If this is the case, go to Step 6.

3. Log in as `acsss`.
4. In a terminal window or at the shell prompt, enter  
`rc.acsss`
5. When the shell prompt returns, log out.
6. Log back in as `acssa` to use the ACSLS software.

This completes second disk configuration.

## Importing the Database

If this is a new installation and an existing database was not exported, skip this section and go to [“Verifying ACSLS Software” on page 430](#).

This section describes how to import and recreate the ACSLS database that you exported in using the `db_export.sh` utility.

**Note:** The procedures outlined below are for importing the database from the default tape device. Refer to “[Utility Reference](#)” on page 241 for instructions on importing the database from other tape devices or from disk.



**CAUTION:** To complete the upgrade, use the `db_import.sh` utility to import the exported database. For more information about using the `db_import.sh` utility.

**To import the database from tape:**

1. Determine whether you are currently the `acsss` user by entering

```
whoami
```

If the response is not `acsss`, log out by pressing `<CTRL>+D`, then log back in as `acsss`.

2. Insert the exported database tape that you exported with the `db_export.sh` command into the tape drive.
3. Run the database `db_import.sh` utility by entering the following at a UNIX prompt.

```
db_import.sh
```

The utility displays several messages as it executes and can take up to 15 minutes to complete.

4. Press `<CTRL>+C` when you see the following message. Do *not* insert a database backup tape.

```
Prepare for database backup ...  
Remove export tape.  
Insert database backup tape.  
[Hit RETURN to Continue or Ctrl-C to exit]
```

## Verifying the Imported Database and Library Configuration

In this procedure you use the `acsss_config` program to verify the database and the library configuration.

**To verify that the database has been imported:**

1. To run the configuration program, enter

```
acsss_config
```

The ACSLS feature and configuration screen appears.

2. Enter 7 to skip library ACSLS feature configuration.

At this point, the system generates a series of messages indicating that the database is being verified.

3. Enter **n** at the prompt to skip library communication software configuration.

Configure library communications? (y or n):

4. Enter **y** at the prompt to build/verify the library configuration.

This step builds a database image of your complete library hardware configuration. Before proceeding, make sure that your library is completely configured, that all subsystems are functional and powered on. Build/Verify library configuration?  
(y or n):

5. Enter **y** when the following prompt displays:

Library server database exists and will be overwritten, continue? (y or n):

As the script builds the library configuration, it displays the following message for each panel in each LSM.

ACS # n, LSM # nn, PANEL # nn, created

The script also generates a library configuration report and appends it to the `/export/home/ACSSS/log/acsss_config.log` file.

6. Enter **n** at the prompt to skip CSI configuration.

Configure client system interfaces?  
(y or n):

7. Back up the database.

Are you backing up the database to tape?

---

**YES** Insert a blank tape when the following message displays:

```
Prepare for database backup...
Insert database backup tape
Press RETURN to continue.
```

The database backup procedure `bdb.acsss` automatically executes and captures the configuration data to enable database journaling.

When the database backup is completed, the following message displays:

```
Database backup successfully
completed.
```

Continue with step 8

---

NO Press <CTRL>+C and then refer to [“bdb.acsss” on page 243](#) to backup the database to disk file, then continue with Step 9.

---

**CAUTION:** The tape drive may appear idle during the backup. It may not make any sounds or flash any lights during this time. **DO NOT** remove the tape from the drive until you see the following success message indicating the backup is complete:

Interrupting the backup process before completion will result in a bad backup.

8. Remove the tape and store it in a safe place.
9. Run `rc.acsss` to start the server.

If the `automatic startup on reboot` option is enabled, ACSLS is initiated automatically when the server system is rebooted.

The ACSLS is now ready for library operations. You may log out and log back in as `acssa`.

## Verifying ACSLS Software

The IVP (Initial Verification Program) is an exerciser that is used to verify that a newly installed library is fully functional. This step is optional. It will mount and dismount specified cartridges to specified drives.

To verify ACSLS, mount and dismount a cartridge by either:

1. Entering mount and dismount commands from a `cmd_proc` (See [“Verifying ACSLS”](#)).
2. Running the Initial Verification Program (IVP) (See [“Running ivp.sh to Verify ACSLS” on page 431](#)).

## Verifying ACSLS

**Mount/dismount a cartridge to verify ACSLS:**

1. Log in as `acssa`.
2. Query the server from the `cmd_proc` by entering  
`query server`

If messages are displayed indicating that the server is in recovery mode, wait for a message indicating that the server is running.

3. If ACSLS is not running, start it log in as `acsss` and start ACSLS by entering

```
rc.acsss
```

4. Do you have at least one cartridge in an LSM?

---

YES Continue with the procedure.

---

NO Enter a cartridge into an LSM. For more information, see “Operator Tasks” in *ACSL System Administrator's Guide* and continue with the procedure.

---

5. Mount a cartridge by entering:

```
mount vol_id drive_id
```

**Hint:** Use the `query drive` command to get the ID of an available drive and the `query volume` command to get the ID of a library cartridge. See “Command Reference” in *ACSL System Administrator's Guide* for more information.

6. Did you see a message indicating a successful mount?

A successful mount message is:

```
Mount: vol_id mounted on drive_id
```

---

YES Procedure is complete, go to step 7.

---

NO If an error message appears, run this verification procedure again, ensuring that you specified a valid, available drive and a library cartridge. If the mount/dismount still fails, call StorageTek for assistance.

---

7. Dismount the cartridge by entering:

```
dismount vol_id drive_id force
```

where `vol_id` is the volume and `drive_id` is the drive you specified in Step 5.

## Running `ivp.sh` to Verify ACSLS

Use this optional test procedure to run the IVP program (`ivp.sh`) to mount and dismount a cartridge. You specify the cartridge and drive that `ivp.sh` uses in the `ivp.dat` file.

**Run `ivp.sh` to mount/dismount a cartridge to verify ACSLS:**

1. Open a Terminal or a Command tool.
2. Change to the directory that contains the `ivp.dat` file.

```
cd /export/home/ACSSS/diag/data
```

3. Using any text editor (for example, `textedit`), open the `ivp.dat` file and enter a drive ID and cartridge ID in the following format at the bottom of the file:

```
drive_id vol_id
```

**Hint:** Use the `query drive` command to get the ID of an available drive and the `query volume` command to get the ID of a library cartridge. See “Command Reference” in *ACSL System Administrator’s Guide* for more information.

4. To start the IVP program, enter

```
/export/home/ACSSS/diag/ivp/ivp.sh
```

`ivp.sh` mounts and dismounts the cartridge that you specified in the `ivp.dat` file in about 30 seconds.

5. Did you see a message indicating a successful mount?

---

YES	Procedure is complete.
NO	If an error message appears, run this verification procedure again, ensuring that you specified a valid, available drive and a library cartridge in the <code>ivp.dat</code> file. If the mount/dismount still fails, call StorageTek for assistance.

---

## ACSL Setup for Client Communication

To set up ACSLS for client communication, you include the client host name and IP address in `etc/hosts` or in the NIS lookup table.

## Regressing to a Previous Version of ACSLS

If for whatever reason you need to regress to a previous version of ACSLS after you install ACSLS 6.1.1, contact Central Software Support (CSS). For more information, see *Requesting Help from Software Support*.

## De-Installing a Second Disk on AIX

To de-install the second disk:

1. If you are not already logged in as `acsss`, log in as `acsss`.
2. From the `cmd_proc` window, idle ACSLS.

```
idle
```

3. Shut down ACSLS by entering

```
kill.acsss
```

4. Shut down the database.

```
db_command stop
```

5. Log in as root.

6. Change directories by entering

```
cd /export/home/ACSSS/install
```

7. To execute the second disk utility, enter

```
./sd_mgr.sh
```

This command starts the second disk manager. With this function, you can remove second disk support.

8. Enter 2 to de-install second disk support when you are prompted for a menu choice.

```
2: De-installation of a second database disk.
```

9. Respond y or n when you see the following prompt:

```
Are you de-installing a bad secondary disk? (y or n):
```

Select y if your second disk is damaged and non-functioning.

Select n if your second disk is not damaged and is functioning.

10. When the Second Disk Support menu appears, select 3 to exit the menu.

A message and prompt display indicating you must do a mandatory backup and to insert a database backup tape.

11. Insert the tape and wait for the backup to complete.

The backup takes several minutes to complete.

12. When the backup is completed, remove the backup tape and store it in a safe place.

13. Log out by selecting Log Out from the Workspace menu.



# ACSL S Backup and Recovery Tools

## B

### ■ ACSLS Backup Tools

ACSL S offers three robust and distinctly different methods for backing up both its database and miscellaneous files. Each utility performs different functions and all methods play an important role in a complete disaster recovery plan.

This appendix:

- Outlines and explains each utility, what they are used for, and why they are important.
- Provides a high level view of disaster recovery scenarios

### Automatic Backups

ACSL S provides automated database protection services. These automated protection services safeguard the daily operation of the ACSLS database against changes that can produce either unintended consequences or from database corruption

As a result of these automated backup protection services, you have the ability to restore your database back to any state from the present to the end of your retention period. The restore tools are discussed later in this appendix.

This section discusses the automated backup methods and why they are used. Refer to [“Automatic Database Backup” on page 215](#).

- ACSLS default backup directory  
During the initial installation of ACSLS, you were asked to supply the name of the directory to use for backups (`/export/backup` by default). It is in this directory that most backup activity occurs.  
By default, ACSLS uses a directory, within this backup directory, called `/export/backup/informix/lib6serv` for most Informix database backup activity.
- When an Informix logical log fills up, it is backed up automatically to a directory called `/export/backup/informixlib6serv/2`.

- When the daily level 0 archive (complete database backup) is performed, this is placed in the directory, `/export/backup/informix/lib6serv/rootdbs`.

The time at which the daily level 0 archives are performed can be modified by changing the “Automatic Backup Variables” within `acsss_config`.

Refer to “*Verifying and Changing Dynamic and Static Variables*” for information on changing default backup behavior.

- Database retention period

Another configurable parameter within ACSLS that affects automatic backups is the database retention period. This is defined as the amount of time ACSLS retains the logical logs and the level 0 archives.

The default for the retention period is 8 days.

## Manual Backups

ACSLS provides a utility called `bdb.acsss` that backs up the critical ACSLS data via the command line. This is also the method used to restore the ACSLS database where the environment consists of the same or identical hardware, OS level, and ACSLS version. Refer to “*Database Backup and Restore*” or the *Utility Reference* appendix for detailed information on the `bdb.acsss` command.

Used without any command line options, `bdb.acsss` provides the ability to create a database backup to a local tape drive to be used for offsite storage. All critical ACSLS database and miscellaneous files are backed up to tape. This tape can then be used to restore ACSLS to its previous state on the same or identical hardware for scenarios such as an internal disk or motherboard failure.

The `bdb.acsss` command also offers two different command line options giving the utility more flexibility.

- Using the `-f` option

Using the `-f` option allows you to specify a tape device other than using the default tape device. This provides you greater flexibility by allowing you to determine where the backup is written. Be careful when writing to a tape device that resides in an ACSLS controlled library because if you lose the database, you lose the location of your recovery tape. It is always a good idea to store a tape of your database backup to an offsite location. At the command line type “`bdb.acsss -f /dev/rmt4`” or some other device to use this option.

Another way to use the `-f` option is to follow the option with a path and file name on disk. This gives you the ability to FTP the file to another server, thereby protecting your backup. With this option, we recommend that you

don't store the backup on the ACSLS server. If there is a disk failure, your backup would be lost. To execute this option, simply type

```
bdb.acsss -f /export/backup/my_backup.bak.
```

- Using the `-s` option

Using `-s` provides a server-to-server option. When you need the flexibility to recreate the ACSLS server environment on different server hardware and/or to a different name, this the option to use. It allows you to create the backup on one server and restore the environment to a different server. Files created using this option, by default, are located in directory `/export/backup/server_to_server` unless a different backup directory was specified at the time ACSLS was installed. When ACSLS is installed, the file `/export/backup/server_to_server/server_to_server.bak` is created. This file is owned by "acsss" and the group is "informix". This file should not be modified because if it is, server to server backups could fail. Refer to ["Backing up ACSLS and Restoring to Another Server" on page 218](#) for detailed information.

## Manual Database Exports

ACSLs provides a utility called `db_export.sh` to export the ACSLS database, miscellaneous ACSLS files, and any customized dynamic variables. The `db_export.sh` utility is responsible for dumping the ACSLS database to comma separated ACSII files, making a copy of the ACSLS miscellaneous files, and making a copy of the dynamic variables. This is the method used to migrate to newer versions of ACSLS and is not recommended for daily backup operations because both ACSLS and the database must be down before performing the export.

The `db_export.sh` command line utility is the preferred method for migrating the database between different levels of server hardware, OS versions, and different releases of ACSLS. Without options, it can be used with the local default tape device such as `/dev/rmt0`. This tape can then be moved to any location, and ACSLS and its associated miscellaneous files can be restored into any OS version or level of ACSLS.

As in the `bdb.acsss` utility, the `-f` option can be used to specify a tape device other than the system default. Simply execute `db_export.sh /dev/rmt4` or any attached tape device to use this option.

The `-f` option also allows the database to be exported to the named file. When using this method, you will notice that two files are created, the one that you named and also another file with a `.misc` extension. Both files must be transferred to the machine where the import will take place to ensure a successful import.

When executing the `db_export.sh` utility either with the `-f` option or without, you will be prompted to choose the version of ACSLS to which you are exporting. Selections are:

- 1: 5.3.2 or 5.4
- 2: 6.0 or 6.0.1
- 3: 6.0.1 with L700e
- 4: 6.0.1 with PUT0201
- 5: 6.1 or later

Choose the appropriate version. Select 5 for all versions greater than 6.1.

For detailed information, refer to [“Exporting to a Disk File” on page 207](#) or [“Exporting to Tape” on page 208](#).

## ■ ACSLS Recovery Tools

ACSLS uses two different recovery tools to restore all backups and exports. Both offer a menu driven user interface and easily selectable options. These two utilities are:

- `rdb.acsss` - the recovery tool for both automated and manual backups
- `db_import.sh` - restores an exported database and/or miscellaneous files from the same version of ACSLS, a different version of ACSLS, or from a different hardware platform

### Using `rdb.acsss`

When typed at the command line, `rdb.acsss` first checks to see if ACSLS and the database are running and if not, it displays a menu. The menu options are:

- Option 1 - Restore from current local disk backup
- Option 2 - Restore from previous local disk backup (to a point-in-time)
- Option 3 - Restore from a previous tape or network file backup
- Option 4 - Restore a backup created on a different server
- Option 5 - Restore ACSLS `non_database` files

#### **Restoring from current local disk backup**

Option 1, “Restore from current local disk backup” is used to recover the database back to its most current state. It first restores the latest level 0 archive and then applies all logical log transactions up until now. This can be used if a database will not initialize.

#### **Restoring from previous local disk backup (to a point-in-time)**

Option 2, “Restore from previous local disk backup (to a point-in-time)” is used to restore the database to any point in time from now back to the retention

period (default of 8 days). When you select this option, all retained backups are displayed and you simply enter the date and time to restore. This is very useful if you make a change with unintended results and need to revert back in time.

## Restoring from a previous tape or network file backup

Option 3, “Restore from a previous tape or network file backup” allows the restoration of a backup file residing on tape or file. In the case of disaster such as disk failure, you would either retrieve the offsite tape or FTP the backup file onto the server and then select this option. It then asks you the location of the file, and restores the database and miscellaneous files to the state that they were in during the backup. This only works if the server hardware is either the same or identical and the name of the server is the same.

## Restoring a backup created on a different server

Option 4, “Restore a backup created on a different server” is used to recreate an ACSLS environment on a different server. In the case of a cold offsite recovery, the name of the server to recover on is different than the server that had the original ACSLS server environment. The Informix installation does not allow itself to be recovered onto a server with a different name from the one on which it was originally installed. The server-to server option takes care of this problem and allows a complete restore of ACSLS and its associated miscellaneous files.

For detailed information, refer to [“Backing up ACSLS and Restoring to Another Server” on page 218](#).

## Restoring ACSLS non-database files

Option 5, “Restore ACSLS non-database files” allows you to restore miscellaneous files without changing the database. This can be useful when configuration files have changed but you are not sure where and when.

## Using `db_import.sh`

ACSL S provides a `db_import.sh` utility to restore an exported database from the same version of ACSLS, a different version of ACSLS, or it could even be from a different hardware platform. Like `rdb.acsss`, it offers an easy to read menu driven user interface allowing you to select the task that you wish to perform.

The `db_import.sh` utility can work without options or you can supply the `-f` option with a path and file name as an argument. Executing `db_import.sh` from the command line without any options causes the utility to look for the exported database on the local tape device. It first checks for the existence of the exported database, verifies that it is a valid database export file, and then displays a menu with four options.

If you provide the `-f` option with a path and file name, `db_import.sh` uses the supplied file name as the exported database file. As with the local tape device, it first checks to see if the file exists and then validates that the supplied file name is an exported database file. If the supplied file is a valid export, it displays a menu. The menu options are:

- Option 1 - Import database only from either a different platform or from an earlier release
- Option 2 - Import miscellaneous configuration files exported from a different platform or from an earlier release
- Option 3 - Recover a previous environment on the same platform and version (Disaster Recovery)
- Option 4 - Merge customized dynamic variables from a previous installation

### **Importing database only from either a different platform or from an earlier release**

Whether you are recovering the database from a local tape device or from a file, choosing Option 1, “Import database only from either a different platform or from an earlier release” initiates an import of the data only and does not recover any miscellaneous files. This is the desirable method for migrating from one platform to another, such as Solaris to AIX, or if you don’t want the miscellaneous files from a previous installation.

### **Importing miscellaneous configuration files exported from a different platform or from an earlier release**

Option 2, “Import miscellaneous configuration files exported from a different platform or from an earlier release” provides the ability to recover miscellaneous files from previous versions of ACSLS or from different platforms. These miscellaneous files would include any access control information or customized volume report templates.

### **Recovering a previous environment on the same platform and version (Disaster Recovery)**

Option 3, “Recover a previous environment on the same platform and version (Disaster Recovery)” is used to recover or rebuild an identical server from the one from which the data was exported. This can be used in situations where you are retiring a server and building a new one. You are able to restore the ACSLS database and miscellaneous files to exactly the same state that was exported.

## Merging customized dynamic variables from a previous installation

Option 4, “Merge customized dynamic variables from a previous installation” gives you the ability to recover customized ACSLS variables from a previous environment. Even if you are unsure if you have any customized variables, it is a good idea to run this option during migration because it will not change any unchanged variables.

## ■ Disaster Scenarios

### Database becomes corrupted – need to restore the database to a working state

1. As user “acsss”, idle and stop ACSLS and the Informix database.

```
>idle      (from cmd_proc)
$ kill.acsss (from UNIX command line)
$ db_command stop (from UNIX command line)
```

2. Start `rdb.acsss` utility
3. Select option 1, “Restore from current local disk backup”
4. Respond with “n” to the following prompt:

```
Are you restoring the primary disk from the second
disk? (This will overlay the current version of the
ACSLs and Informix configuration files.) (y or n):
```

5. When the recovery is complete, start ACSLS and test

### Ran `acsss_config` against the wrong library

1. As user “acsss”, idle and stop ACSLS and the Informix database.
 

```
>idle      (from cmd_proc)
$ kill.acsss (from UNIX command line)
$ db_command stop (from UNIX command line)
```
2. Start `rdb.acsss` utility
3. Select option 2, “Restore from previous local disk backup (to a point in time)”
4. Enter the date and time to restore (just a couple of minutes before the bad configuration)
5. Start ACSLS and test

Primary Disk failure – no second disk installed

1. Replace the failed disk according to the manufacturer's instructions.
2. Install the operating system.

Refer to the following chapters:

- Solaris – step 3
- AIX all versions – Manufacturer provided instructions

3. Install ACSLS.

Refer to the following chapters:

- Solaris – steps 4 and 5
- AIX – steps 5 and 6

4. Insert tape with backup or FTP backup file onto the server.
5. Start `rdb.acsss` utility
6. Select option 3 “Restore from a previous tape or network file backup”
7. Enter the path to either the file or tape device.  
Leave this blank and press Enter if the backup is on the local tape device.
8. Exit the `rdb.acsss` utility
9. Start ACSLS and test according to procedures in “*Database Backup and Restore*” chapter.

## **Disk failure on primary disk with /export/home – second disk support installed**

1. Replace the failed disk according to the manufacturer's instructions.
2. Install the operating system.

Refer to the following chapters:

- Solaris – step 3
- AIX all versions – Manufacturer provided instructions

3. Install ACSLS.

Refer to the following chapters:

- Solaris – steps 4 and 5
- AIX – steps 5 and 6

4. Enter the `rdb.acsss` recovery utility.

5. Select option 1, “Restore from current local disk backup.
6. Respond with “y” to the following prompt:
 

```
Are you restoring the primary disk from the second
disk? (This will overlay the current version of the
ACSLs and Informix configuration files.) (y or n):
```
7. Either accept the default location for `second_disk` support (`/second_disk`) or enter the location where second disk was installed for your system.
 

The database recovery occurs. You are then prompted to perform a backup to tape.
8. Responding `no` will still initiate a backup to local disk.
9. Start ACSLS and test following the procedures in “*Database Backup and Recovery*” chapter.

## Disk failure on `/second_disk` – second disk installed

1. Replace the failed disk according to the manufacturer’s instructions.
2. As root, run `$ACS_HOME/install/sd_mgr.sh` to re-install `second_disk` support.
 

Refer to the following chapters:

  - Solaris – step 10
  - AIX – step 11
3. Start ACSLS and test following the procedures in *Database Backup and Recovery*.

## Server failure – rebuilding the same server with new hardware

1. Install the operating system.
 

Refer to the following chapters:

  - Solaris – step 3
  - AIX all versions – Manufacturer provided instructions
2. Configure the new server and OS with the settings from the previous server.
 

It is important that the name and IP address of this server be the same as the old server.
3. Install ACSLS.

Refer to the following chapters:

- Solaris – steps 4 and 5
  - AIX – steps 5 and 6
4. Insert the backup tape or FTP backup file onto the server.
  5. Start the `rdb.acsss` utility
  6. Select option 3 “Restore from a previous tape or network file backup”
  7. Enter the path to either the file or tape device.  
Leave this blank and press Enter if the backup is on the local tape device.
  8. Exit the `rdb.acsss` utility
  9. Start ACSLS and test according to procedures in *Database Backup and Restore*.

## Server failure – rebuilding another ACSLS server with new hardware

1. Install the operating system.  
Refer to the following chapters:
  - Solaris – step 3
  - AIX all versions – Manufacturer provided instructions
2. Install ACSLS.  
Refer to the following chapters:
  - Solaris – steps 4 and 5
  - AIX – steps 5 and 6
3. Place the server-to-server backup files in the proper location.
4. Enter `rdb.acsss`.
5. Select option 4, “Restore a backup created on a different server”.
6. When the recovery utility completes, start ACSLS and test according to procedures in *Database Backup and Restore*.

# ACSL S Support of the SL8500

---

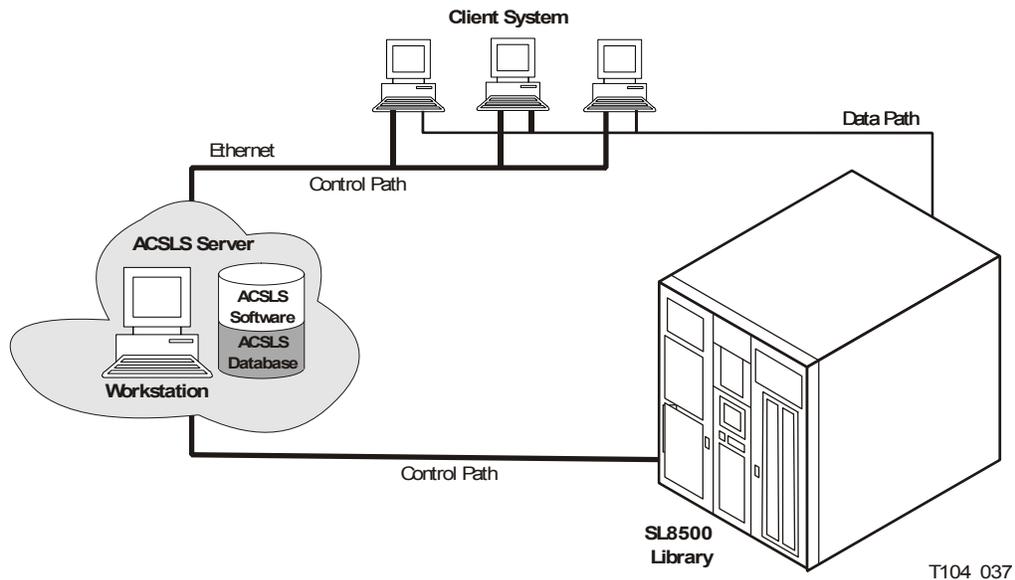


This appendix discusses the following topics:

- [“Connecting the SL8500 to ACSLS” on page 446](#)
- [“Before Configuring ACSLS for the SL8500” on page 446](#)
- [“Understanding SL8500 Internal Addresses and ACSLS Addresses” on page 447](#)
- [“Using SLConsole to Translate Addresses” on page 449](#)
- [“SL8500 CAP Behavior” on page 453](#)
- [“Minimizing Elevator and PTP Activity” on page 454](#)
- [“Configuring Tape Drives to Support Your Workloads” on page 455](#)
- [“Managing Cartridge Locations” on page 456](#)
- [“Finding Missing Cartridges” on page 457](#)
- [“Varying the SL8500 Offline” on page 457](#)
- [“Using the Dynamic Configuration \(config\) utility” on page 460](#)
- [“Work around when an inoperative HandBot is adjacent to the middle SL8500 CAP magazine” on page 461](#)
- [“Using ACSLS HA” on page 462](#)
- [“Auditing the SL8500 After Expansion” on page 463](#)
- [“Connecting SL8500s with Pass-Thru-Ports” on page 466](#)
- [“Updating ACSLS After Adding HandBots” on page 477](#)
- [“ACSL S Dual TCP/IP Support on SL8500” on page 478](#)

The following figure shows the SL8500 library with an ACSLS server.

**Figure 50. SL8500 with an ACSLS Server**



## ■ Before Configuring ACSLS for the SL8500

Before you configure ACSLS for the SL8500:

- Connect the SL8500 to ACSLS
- Verify that all the components of the SL8500 are operational.

ACSLS builds its library configuration from the information reported by the library. If SL8500 components are not operational, the library information may not be reported to ACSLS, and the ACSLS configuration of the SL8500 will be incomplete.

## Connecting the SL8500 to ACSLS

The SL8500 library uses TCP/IP protocol over an Ethernet physical interface to manage and communicate with the host and ACSLS. This interface enables ACSLS to connect to and communicate with the SL8500.

- All hosts should communicate with one and only one SL8500 library.

This is critical when creating an SL8500 Library Complex, which is done by connecting libraries together with pass-thru ports. All hosts must connect to only one library in the complex and preferably to the first or rightmost library in the complex.

Making multiple “host” communication connections to more than one library will cause problems.

- It is preferable to have the SL8500 library on a separate subnet or a controlled network to protect it from floods of ARP broadcasts.

For more information, refer to the *SL8500 Modular Library System Technical Brief - Host to Library Communications*.

## Verifying that all SL8500 Components are Operational

To verify that all the components of the SL8500 are operational:

1. Logon to the StreamLine Library Console™ (SLConsole).  
You can use either the console on the SL8500 or a remote Library Console.
2. Select `Tools -> System Detail`.
  - All SL8500 components should be green.  
**Exception:** Drives that are yellow can be configured now, or later, using dynamic configuration (“[config drives](#)” on page 249).
  - Missing components can be added using the Dynamic Configuration (`config acs` or `config lsm`) utility.
  - **IMPORTANT:** Before configuring the SL8500, the elevators (Elevator Folder) **must** be green. If the elevators are not green, **do not** configure the SL8500 to ACSLS. The elevators are the logical pass-thru-ports (PTPs). Without PTPs, ACSLS will not know that the SL8500 rails are connected.
3. Once the SL8500 components are operational, configure SL8500 to ACSLS, as discussed in “[Configuring or Updating your Library Hardware](#)” on page 130 or under the “[Dynamic Configuration \(config\) utilities](#)” on page 245.

## ■ Understanding SL8500 Internal Addresses and ACSLS Addresses

There are differences in the internal addresses of the SL8500 and other libraries supported by ACSLS and HSC.

- The SL8500 is *one’s*-based (1) and uses negative numbers.
- Other libraries use a *zero*-based (0) with *no* negative numbers.
- The SL8500 uses *five* parameters: library, rail, column, side, and row.
- Other libraries use: ACS, LSM, panel, row, and column (HLI-PRC).

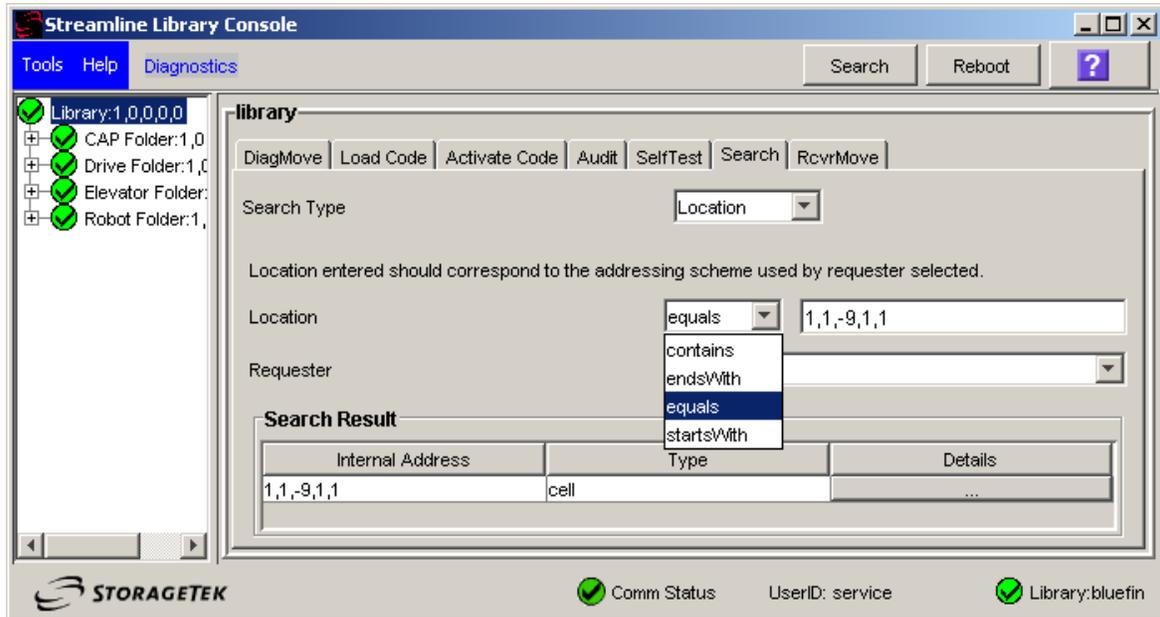
**Table 33. Addressing Descriptions**

HLI-PRC	SL8500	Description	
<b>ACS</b>	<b>Library</b>	Number of the specific SL8500 library in a library complex. An ACS is a SL8600 library complex. There can be multiple SL8500s in a library complex.	
<b>LSM</b> LSM 0 ⇨ LSM 1 ⇨ LSM 2 ⇨ LSM 3 ⇨	<b>Rail</b> Rail 1 Rail 2 Rail 3 Rail 4	The SL8500 library has four <i>rails</i> that the HandBots travel, which are numbered from top to bottom <b>1–4</b> (one’s-based).  ACSL S considers <i>each rail</i> to be a separate <i>LSM</i> , numbered from top to bottom <b>0–3</b> (zero-based).	
<b>Panel</b> Panel 0 ⇨ Panel 1 ⇨ Panel 2– <i>n</i> ⇨	<b>Column</b> CAP Drives Storage slots	Columns indicate the <i>horizontal</i> location in the library. As viewed from the front of the library column and panel numbers start at the <i>center</i> of the drive panel (1) and sweep forward with increasing numbers.  <b>Note:</b> The SL8500 does not use panels as an address.  An HLI panel spans across the width of the library to include both sides (left and right) and both walls (inner and outer) for <i>each</i> LSM.	
	<b>Side</b>	Wall location  1. Outer wall 2. Inner wall	HandBot number  1. Left (–) 2. Right (+)
<b>Row</b>	<b>Row</b>	Rows indicate the <i>vertical</i> location of a tape cartridge and are numbered from the top—down.	
<b>Column</b>		Rows for the <b>HLI</b> address are: <ul style="list-style-type: none"><li>Storage panels start at 2 with Column 0 = left and Column 1 = right</li><li>Rows <b>0–12</b> outer walls</li><li>Rows 13–26 inner walls</li><li>Each column in a normal storage panel has 27 rows.</li><li>For a total capacity of 54 cartridges per panel.</li></ul>	Rows for the <b>SL8500</b> address are: <ul style="list-style-type: none"><li>Storage slots start at Column -3 = left Column +3 = right</li><li>Rows <b>1–13</b> outer wall</li><li>Rows <b>1–14</b> inner wall</li></ul>
<ul style="list-style-type: none"> <li>Zero-based numbering (as with HLI) starts numbering at 0.</li> <li>One’s-based numbering (as with the SL8500) starts numbering at 1.</li> <li>This is an important difference in the numbering sequences between software (ACSL S or HSC) and hardware (physical SL8500 addresses)</li> </ul>			

## ■ Using SLConsole to Translate Addresses

You can use the StreamLine Library Console™ (SLConsole) Search utility to translate between the SL8500 internal address and the ACSLS or HSC panel, row, and column. To locate a cartridge:

1. Log in to the SLConsole.
2. Select Diagnostics -> Search.



3. Select Location.
4. Select one of the following operations in the Location field:

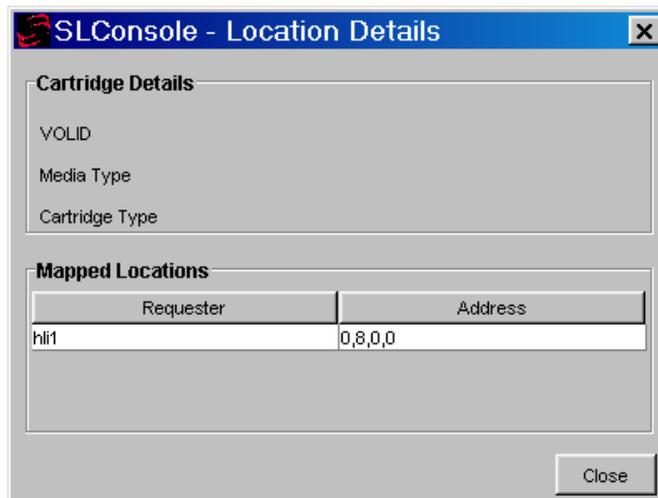
contains	Example: 1,1,-9 lists the contents in Library 1, Rail 1, Column -9 for all rows on both sides
endsWith	Example: 1,5 lists the slot contents for all rails and columns for Side 1, Row 5
equals	Example: 1,1,-9,1,1 lists the contents in that specific location (L,R,C,S,W)
startsWith	Example: 1, 3 lists the slot contents for all columns, sides, and rows in Library 1, Rail 3

5. Select one of the following from the Requestor pull-down menu.

- default

The physical location inside the library (cell, drive, CAP).

If you know the physical location (the internal address), and need to find the HLI-PRC address, enter that address in the *location* and pick *default* as the requester.

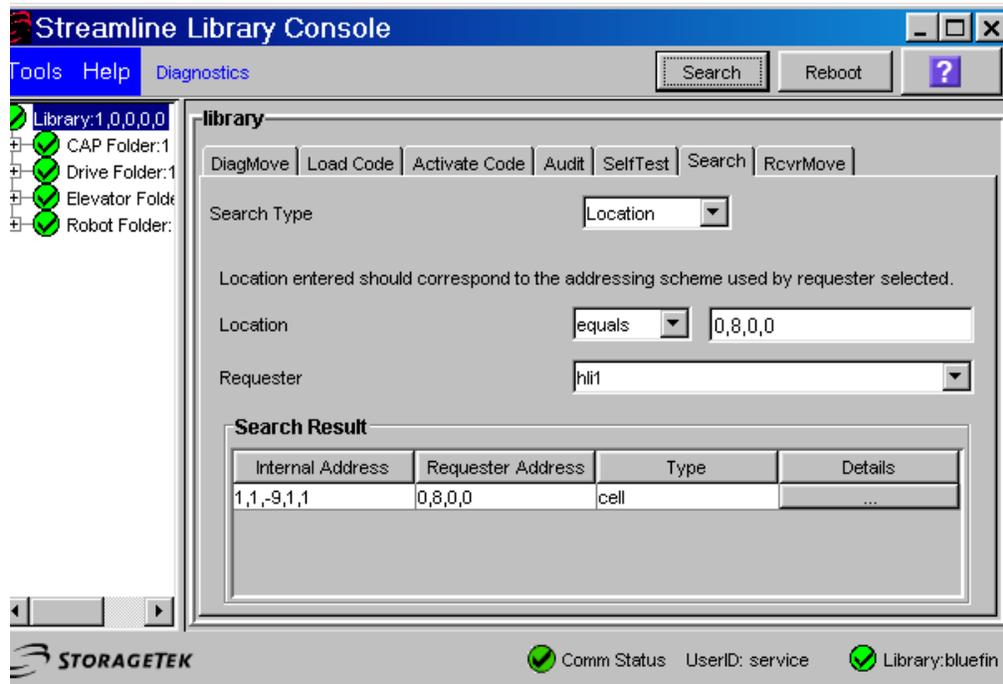


- hli1

This is the HLI-PRC address of the cartridge from the library management software.

If you know the HLI-PRC address and want to find the physical location (internal address) enter that address in the *location* and pick *hli1* as the requester.

This option displays both the internal address and the hli1 Requester address as shown in the following example.



6. Click the Search button in the top right corner of the SLConsole.  
The search result lists the location by slot-type (cell, drive, or CAP).
7. Click the *Details* button for more information such as: VOLID: media and cartridge type (LTO, SDLT tape, and T-Series; data, cleaning, or diagnostic); and HLI address for cartridges when you specify a *default Requester*.

## Tape Drive Locations

The tape drives are associated with and belong to an *LSM*. To mount a cartridge tape in a different LSM, the cartridge must go through an internal pass-thru operation (in this case, *the elevator*) to the drive.

The following tables shows the **internal**—software—mapping (viewed from looking *inside* the library at the tape drives), and

the **external**—physical—numbering of the drives (looking *outside* at the rear of the Drive and Electronics Module).

**Table 34. Software Drive Numbering**

**Table 35. Physical Drive Numbering**

LSM		ACSL S — Drives Numbers			
Rail 1	LSM 0	<b>Drive 0</b>	Drive 4	Drive 8	Drive 12
		Drive 1	Drive 5	Drive 9	Drive 13
		Drive 2	Drive 6	Drive 10	Drive 14
		Drive 3	Drive 7	Drive 11	Drive 15
Rail 2	LSM 1	Drive 0	Drive 4	Drive 8	Drive 12
		Drive 1	Drive 5	Drive 9	Drive 13
		Drive 2	Drive 6	Drive 10	Drive 14
		Drive 3	Drive 7	Drive 11	<b>Drive 15</b>
Rail 3	LSM 2	Drive 0	Drive 4	Drive 8	Drive 12
		Drive 1	Drive 5	Drive 9	Drive 13
		Drive 2	Drive 6	Drive 10	Drive 14
		<b>Drive 3</b>	Drive 7	Drive 11	Drive 15
Rail 4	LSM 3	Drive 0	Drive 4	Drive 8	Drive 12
		Drive 1	Drive 5	Drive 9	Drive 13
		Drive 2	Drive 6	Drive 10	Drive 14
		<b>Drive 3</b>	<b>Drive 7</b>	<b>Drive 11</b>	<b>Drive 15</b>

External—Physical Drive Numbers			
Drive 61	Drive 62	Drive 63	<b>Drive 64</b>
Drive 57	Drive 58	Drive 59	Drive 60
Drive 53	Drive 54	Drive 55	Drive 56
Drive 49	Drive 50	Drive 51	Drive 52
Drive 45	Drive 46	Drive 47	Drive 48
Drive 41	Drive 42	Drive 43	Drive 44
Drive 37	Drive 38	Drive 39	Drive 40
<b>Drive 33</b>	Drive 34	Drive 35	Drive 36
Drive 29	Drive 30	Drive 31	Drive 32
Drive 25	Drive 26	Drive 27	Drive 28
Drive 21	Drive 22	Drive 23	Drive 24
Drive 17	Drive 18	Drive 19	<b>Drive 20</b>
Drive 13	Drive 14	Drive 15	Drive 16
Drive 9	Drive 10	Drive 11	Drive 12
Drive 5	Drive 6	Drive 7	Drive 8
<b>Drive 1</b>	<b>Drive 2</b>	<b>Drive 3</b>	<b>Drive 4</b>

These tables show a matching of drives (the highlighted drives). *For example:*

- Internal/software LSM 0 Drive 0 matches with external/physical Drive 64.
- Internal LSM 1 Drive 15 matches with external/physical Drive 33.
- Internal LSM 2 Drive 3 matches with external physical Drive 20.

## ■ SL8500 CAP Behavior

SL8500 CAPs span three rails (2-4). For ACSLS, this is LSMs 1-3. The SL8500 has one CAP with an optional second CAP. Each CAP has three magazines with 13 cells per magazine. Each magazine is adjacent to a separate rail, and can only be accessed by the handbots on that rail.

Because the second CAP is not present, ACSLS reports that the CAP is offline or non-operational. To prevent any messages regarding the CAP from being displayed, *vary* the non-existent CAP to an offline state using the *vary* command. This also prevents any mistaken attempts to use the second CAP where it does not exist.

The CAP is treated as an independent component, and does not belong to any one rail (LSM).

### Examples:

- The SL8500 CAP can be online and operational even though the LSM identified in the CAP ID is offline or not operational.
- If LSM 1 is varied offline, the SL8500 CAP(s) are not automatically varied offline.
- If LSM 1 is varied online, ACSLS will not automatically vary the SL8500 CAP(s) online.
- The SL8500 CAP(s) can be online and used to enter and eject cartridges to and from SL8500 LSM 0, even though some of the SL8500 LSMs 1-3 are offline and/or not operational.

**Note:** For CAP behavior in a partitioned library, refer to [“CAP Behavior” on page 494](#).

## Enter or Eject Operations

During an enter, ACSLS will always try to move the cartridge to an LSM (rail) adjacent to the CAP magazine. For ejects, ACSLS will always try to eject cartridges to a CAP cell adjacent to the LSM containing the cartridge

If these two operations are not possible, the library controller takes care of moving the cartridge through the elevator to another LSM. This requires movements between two handbots and the elevator.

## ■ Minimizing Elevator and PTP Activity

There are several things you can do to minimize elevator and PTP activity, such as:

- |                            |   |
|----------------------------|---|
| <b>Mounting cartridges</b> | <p>Whenever possible when mounting a tape, use cartridges and tape drives that are in the same LSM.</p> <p>LSM refers to a single rail within the SL8500 library. Each SL8500 contains 4 LSMs.</p>  |
| <b>Using float</b>         | <p>Take advantage of the ACSLS “float” option (enabled by default by ACSLS) by maintaining some free cells within each LSM. Cartridge float is a feature that allows ACSLS to place a dismantled tape cartridge in an empty slot in the same LSM or a closer LSM to the tape drive if the tape originally came from a different LSM using a pass-thru operation.</p> <p>When cartridges are dismantled, ACSLS tries to avoid elevator (pass-thru) activity among LSMs by assigning a new home cell whenever the cartridge's old home cell is in a different LSM. ACSLS attempts to put the cartridge away:</p> <ul style="list-style-type: none"> <li>• in the same LSM as the tape drive from which it was dismantled</li> <li>• or to the closest LSM (with free storage cells) to the drive</li> </ul> |
| <b>Entering cartridges</b> | <p>Enter cartridges into an LSM that has compatible tape drives for the media being entered.</p> <p>Example: You have only LTO drives on LSMs 2 and 3, and you want the LTO cartridges to be located in these LSMs. When entering these cartridges, you should place them in the CAP magazines adjacent to LSMs 2 and 3. ACSLS then makes every effort to put the cartridges in the LSM that is adjacent to that CAP magazine.</p>  |
| <b>Scratch cartridges</b>  | <p>Make sure that scratch cartridges are available in sufficient quantity in each LSM where they will be used. For an SL8500, this means having scratch cartridges available on each rail (LSM) of the library.</p>   |
| <b>Free cells</b>          | <p>Make sure there are adequate free cells in each LSM.</p>   |

## ■ Configuring Tape Drives to Support Your Workloads

How tape drives are configured in the SL8500 can minimize both elevator and PTP activity while supporting your tape workloads. Strategies to use in determining where tape drives are located in the SL8500 include:

- Cluster cartridges by workload, with enough drives to support the maximum drives needed for the workload. Separate the cartridges used by each workload on separate rails, and ensure the rail(s) dedicated to a workload has enough drives to meet the maximum concurrent mounts for the peak usage of the workload. Ensure that the rail has not only the tape cartridges for the workload, but also the scratch cartridges that will be needed.
- Allocate separate rail(s) to each major application workload. For example: separate Veritas NetBackup and Tivoli applications all need media and drives.
- Clustering drives and media on a single rail works until the mounts per hour threshold is reached, all drives are in use, or there are too many active cartridges to fit on a rail. When the resources needed for a workload exceeds the capacity of a rail, spread the cartridges and drives over two or more rails.
- Cluster drives by type, placing drives that use different media types on separate rails (LSMs). For example, place T9840 drives on one rail and T9940 drives on a different rail.
- Configure your heavy tape applications so they will not exceed the performance limits of your library configuration.
- Configure the SL8500 with 8 HandBots (two HandBots per rail) to provide redundancy. This allows you to always access the cartridges and drives that support a workload.

## ■ Managing Cartridge Locations

How cartridges are originally entered in the library or their status in the library can have an affect on ACSLS performance. Considerations are:

**Entering cartridges** *Recommendation:* Enter cartridges through the CAP.

When manually placing cartridges in the library with the front access door open, *library operations cease* and ACSLS must perform a *full audit* to update the library database to match the actual contents of the library.

*To maximize performance:* Enter cartridges through the cartridge access port (CAP).

During an enter, *the library stays online, mounts can continue*, and the library management software always tries to move the cartridge to an LSM adjacent to the CAP magazine—minimizing pass-thru activity.

If this is not possible, the library controller moves the cartridge through the elevator to another LSM—which requires additional movement between two HandBots and the elevator.

**Clustering cartridges**

Cluster cartridges by workload on separate rails with enough tape drives to support the maximum activity—peak usage—for that workload.

**Using float**

*Recommendation:*

When *float* is on, ACSLS selects a new home cell for a cartridge that is in an LSM as close to the drive as possible on a dismount. This option automatically clusters cartridges by the drives for the workload.

Make sure each LSM contains enough free cells to allow selection of a new home cell in that LSM.

**Supplying scratch cartridges**

Make sure each rail has the correct amount and type of data cartridges *plus* enough scratch cartridges to support the workload.

## ■ Finding Missing Cartridges

If a cartridge is out of place or unaccounted for by ACSLS:

1. Perform a physical audit of the SL8500 using the SLConsole.

The physical audit of the SL8500 is performed as a background task in between handling mount and other library operation requests.

**Warning:** If the SL8500 contents are out of sync with ACSLS due to manual operations such as loading cartridges directly, it is not advisable to attempt continued operations.

If you want to manually add tapes, adding them to a particular LSM within the SL8500 is a better approach. Adding tapes to a particular LSM and auditing only the affected LSM is a quicker and more reliable solution.

To do this, vary the affected LSM to a diagnostic state to ACSLS while the audit is in process. After the SL8500 library audit is performed vary the LSM online to ACSLS.

2. Run an ACSLS audit to update the ACSLS database to match the actual inventory of library cartridges.

## ■ Varying the SL8500 Offline

You should vary SL8500 components offline to ACSLS before they are powered off, if they are inoperative, and before you open an SL8500 access door. This notifies ACSLS that they are unavailable. Once they are available, vary them back online.

### Use ACSLS to vary SL8500 components offline, not SLConsole

You should vary SL8500 components (ACSSs, LSMs, and CAPs) offline to ACSLS, not the SLConsole.

- ACSLS allows outstanding requests to complete before taking components offline, unless it is a vary offline force. The SLConsole has no knowledge of outstanding requests to ACSLS.

Varying components offline using SLConsole may cause requests in progress to fail.

- Prior to SL8500 firmware version 2.52, the “LSM Not Ready” (offline) status is not communicated to ACSLS.

In this case, you must vary components offline to ACSLS.

## When to vary SL8500 components offline to ACSLS

### ***Before opening the access door***

Before opening the SL8500 access door vary the ACS or all four LSMs offline.

- For a standalone SL8500, vary the ACS offline:

```
vary acs acs_id offline
```

- For a SL8500 connected through PTPs, vary all four LSMs (in the SL8500 whose access door will be opened) offline using the following command four times (once for each of the four LSMs):

```
vary lsm lsm_id offline
```

### ***If an LSM (rail) is inoperative***

Prior to SL8500 firmware version 2.52, the “LSM Not Ready” (offline) status is not communicated to ACSLS.

In this case you must vary components offline to ACSLS with:

```
vary lsm lsm-id offline
```

With version 2.52, the SL8500 library notifies ACSLS that the LSM is not ready (inoperative).

### ***If a CAP is inoperative***

If the CAP is inoperative, vary it offline:

```
vary cap cap_id offline
```

## When closing the Service Safety Door

Whenever replacing hardware requires using the Service Safety Door, it is advisable to keep that Service Safety Door closed for the minimum amount of time possible. This is because the Service Safety Door blocks other hardware components (elevators, CAPs, and cells) to which access may be required for completing specific requests.

- Before closing the Service Safety Door on either the left or right side of the SL8500, vary the elevator on that side offline through the SLConsole.

After the Service Safety Door is opened, vary the elevator on that side back online through the SLConsole.

- When the Service Safety Door is closed on the right side, it will block access to the CAP.
  - Before closing the Service Safety Door on the right side of the SL8500, vary the CAP offline through ACSLS.

- After the Service Safety Door is opened, vary the CAP online through ACSLS.

**Note:** When the SL8500 Service Safety Door is closed to separate a service bay from the rest of the library, the CSE can open the access door on that side without taking the LSM or ACS offline.

## When using the Service Safety Door do not use these ACSLS commands and utilities

There are some ACSLS commands and utilities that should not be in progress or initiated when the Service Safety Door is being used. These commands are:

**When the Service Safety Door is closed on either side, do not use these utilities:**

- `acsss_config`
- `config`

**When the Service Safety Door is closed on the right (CAP) side, do not use these commands:**

- `enter`
- `eject`
- `set cap mode auto <cap_id>`

**When the Service Safety Door is closed on the right (CAP) side, the following commands can be used, but special considerations apply:**

- `audit`

The `audit` command can be used. However, if there is a need to eject cartridges as a result of the audit (e.g. because the audit encounters duplicates or unreadable labels), the audit will complete and update the ACSLS database, but the cartridges will **not** be ejected.

- `vary acs` and `vary lsm`

These commands will succeed, but messages will be displayed on `cmd_proc` and the event log reporting CAP failures and inoperative CAPs.

## ■ Using the Dynamic Configuration (config) utility

The dynamic configuration (config) utility allows you to implement configuration changes to ACSLS libraries (and components) while ACSLS remains online and running. These configuration changes are recorded in the `acsss_config.log` file.

The following dynamic configuration utilities are supported:

- `config acs`
- `config drives`
- `config lsm`
- `config ports`

Using the `config` utility provides the following benefits:

- ACSLS can continue running, allowing you to perform mount requests to unaffected library components.
- Allows you to reconfigure specified library components while all other configuration information remains unchanged. For example, when specifying:
  - An ACS, the configurations of other ACSs are not affected.
  - An LSM, the configurations of other LSMs are not affected.
  - A drive panel, the drives on a panel, mounts and dismounts to all existing drives are not affected.

**CAUTION: If you have not installed PUT0601, Do NOT add SL8500 libraries with more than one Storage Expansion Module using dynamic configuration.**

To add these libraries, you must:

1. Stop ACSLS
2. Run `acsss_config` to update the ACSLS configuration.
3. Start ACSLS.

## ■ Work around when an inoperative HandBot is adjacent to the middle SL8500 CAP magazine

This work around is only needed if you have not installed PUT0601 to ACSLS and SL8500 firmware 3.02 or later. With this ACSLS and SL8500 software installed, ACSLS and the SL8500 will use the CAP (except the middle CAP magazine cannot be accessed).

Currently, in an SL8500 when the HandBot adjacent to the middle CAP magazine is inoperative, you cannot use the CAP. This causes all enter and eject requests from ACSLS to fail.

The middle CAP magazine is adjacent to the third SL8500 rail. On a single SL8500, this is LSM 2.

The middle CAP magazine can be inaccessible on:

- a four HandBot SL8500, when the only robot on LSM 2 is inoperative
- an eight HandBot SL8500, when the robot closest to the middle CAP magazine is inoperative. This is the right HandBot in a dual HandBot configuration.

The following work around allows you use the SL8500 CAP when the HandBot adjacent to the middle magazine is inoperative:

1. Start an enter through ACSLS.
2. Open the CAP and remove the bottom magazine.

This leaves the top two magazines in the CAP, but the second magazine cannot be accessed. Only the top magazine can be used for enters and ejects.

- To enter cartridges:
  - Place them in the top magazine and close the CAP.
  - Continue entering cartridges using only the top magazine.
- To eject cartridges:
  - Leave the top magazine empty, close the CAP, and terminate the enter.
  - Eject cartridges - ACSLS will place cartridges only in the top magazine.

Do **NOT** place the bottom magazine back into the CAP until the robot adjacent to the middle CAP magazine is operational.

## ■ Using ACSLS HA

ACSL S High Availability (HA) is a software/solution offered by Professional Services. This solution is available for the Solaris SPARC 9 and 10 platforms.

In environments where there is only one ACS, it is sometimes desirable to fail over highly available servers in the event that communication is lost to the library. The ACSLS HA agent contains a variable that causes different behaviors based on these environmental considerations called "FAIL\_OVER". This variable exists within the script located in `/opt/VRTSvcs/bin/STKLMU/monitor`, and can contain one of two values, "0" or "1".

- Setting the variable to "0" prevents failover
- Setting the variable to 1
  - You can have the primary ACSLS server automatically fail over to the standby server in an attempt to resolve library communication failures.

**CAUTION:** Setting this variable to "1" is NOT recommended in environments with multiple libraries. If this is set in environments with multiple libraries, failing over will impact ALL libraries.

- You can have the primary ACSLS server automatically fail over to the standby server in an attempt to resolve library communication failures.

**Note:** Please refer to the Readme for PTF830785 for a complete description of the FAIL\_OVER variable

## ■ Auditing the SL8500 After Expansion

Expansion occurs when Storage Expansion Modules are added to the SL8500 to increase its capacity. When this happens the Customer Interface Module (CIM), which includes the CAP, must move out. As a result, newer and higher panel numbers are assigned to the three cell panels on the CIM.

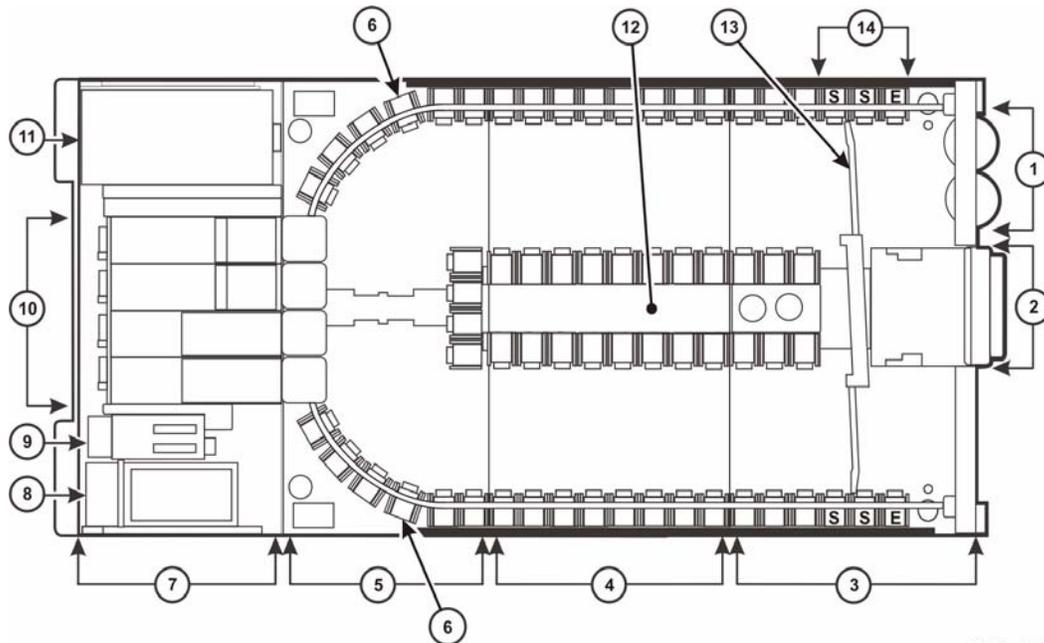
When the cell panels on the CIM are assigned higher panel numbers, the addresses of all the cartridges on the CIM change. You must audit these panels so ACSLS can update its database with the new addresses of these cartridges.

**Note:** If you have to remove any cartridges to attach the Storage Expansion Module, place them in an empty panel in the Storage Expansion Module. Then audit this panel after auditing the three highest panels in the CIM. This guarantees audit will find these cartridges in their new address.

### **Example:**

In the following diagram, the three cell panels in the Customer Expansion Module (# 4) are the three columns of cartridges on both the outside and inside walls on the drive side of the CEM. These are the only cell addresses in the CIM that HandBots can access when the service safety door is activated.

**Figure 51. SL8500 Customer Expansion Module**



L203\_054

- |  |  |
|--|--|
| 1. Cartridge Access Ports (2) Caps           | 8. AC Power Supplies Electronic Control Module |
| 2. Facade Operator Panel (Optional) Icey Pad | 9. DC Power Supplies                           |
| 3. Customer Interface Module                 | 10. Tape Drives                                |
| 4. Storage Expansion Module                  | 11. Accessory Racks                            |
| 5. Robotics Interface Module                 | 12. Inner Wall                                 |
| 6. PTP (Pass Through Port)                   | 13. Service Door                               |
| 7. Drive Electronics Module                  | 14. Reserved Columns                           |
|  | E= End Stop                                    |
|  | S= System Cell                                 |

## Auditing New Cartridge Addresses

When auditing the SL8500 after an expansion:

1. First, find and update cartridges whose addresses have changed by auditing the three highest panels in each LSM (rail).
  - When audit finds a cartridge in a new address, it checks the old address where it was previously recorded in the ACSLS database to verify that this is not a duplicate cartridge.
  - If a cartridge has changed addresses, ACSLS updates its database with the cartridge's new address.

**WARNING:** If ACSLS first audits old cartridge addresses after an expansion, it will not find the cartridges and will delete them or mark them absent. If the cartridges are marked absent, they will be reactivated later when ACSLS audits their new address. However, if

**the cartridge is marked as scratch status, it will lose that status. If the cartridges are deleted, information such as locks, ownership, cleaning cartridge usage, scratch status, and pool assignments are all lost.**

2. Last, audit the entire LSM.

When you audit the entire LSM *last*, you do not lose any cartridge information.

## Procedure for Auditing an Expanded SL8500

After an expansion has occurred

1. Vary the LSMs to a diagnostic state until the configuration is updated and cartridge addresses have been updated with audit.
2. Backup the ACSLS database.
3. Update the configuration in the ACSLS database using one of the following.

- Dynamically, if PUT0501 to ACSLS 7.1 (or later) maintenance is installed:

**config acs *acs\_id***

or

**config lsm *lsm\_id*** (for each LSM (rail) in the SL8500)

- Statically, if Dynamic Configuration is not installed:
  - a. Shutdown ACSLS: **kill.acsss**
  - b. Update the ACSLS configuration: **acsss\_config**
  - c. Bring up ACSLS: **rc.acsss**
- 4. Determine the panel numbers for the three panels on the Customer Interface Module.

If you placed cartridges that were removed to install the Storage Expansion Module(s) in a new panel, determine this panel number.

**display panel *acs\_id,lsm\_id,\****

Where:

***acs\_id*** is the ACS.

***lsm\_id*** is one of the LSMs. All rails (LSMs) on the SL8500 will have the same number of panels

\* displays all panels in the LSM.

5. First, audit the three panels with the highest panel numbers first in each LSM (rail) in the expanded SL8500.

Then, audit the panel(s) where cartridges that were removed during the installation of the Storage Expansion Module(s) were placed.

6. Vary the LSMs online.

Auditing the panels, above, should have updated the addresses for all cartridges that moved.

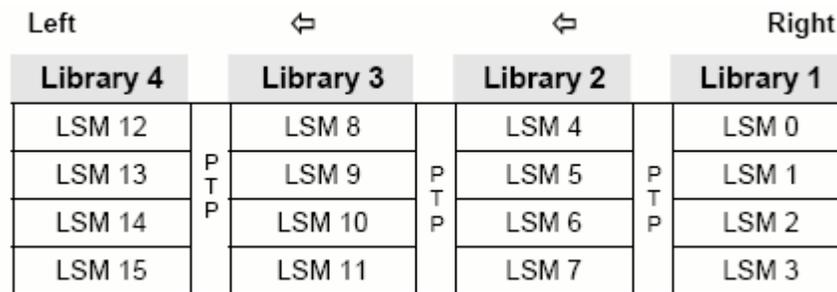
7. Audit each entire LSM in the expanded SL8500.
8. Backup the ACSLS database.

## ■ Connecting SL8500s with Pass-Thru-Ports

You may need to audit SL8500(s) to update cartridge addresses when adding SL8500 libraries to an existing SL8500 library complex.

LSMs in the SL8500 complex are numbered from top down and right to left when viewed from the CAP end, as shown in the following figure.

**Figure 52. Four connected SL8500 Libraries**



Depending on how your site is set up, you may be adding a new SL8500 to the right or left of an existing SL8500. As a result:

- When a new SL8500 is added on the left, the new LSMs have higher numbers, and existing LSMs are not re-numbered. The disruption is minimal.
- When a new SL8500 is added on the right, all existing LSMs are re-numbered. This causes the home cell addresses of all existing cartridges to change.

## Before You Install SL8500 PTP Connections

1. Apply all relevant maintenance to ACSLS before installing SL8500 connections.
2. Enable absent cartridge support.

When absent cartridge support is enabled, ACSLS will not delete cartridges in an LSM that is deleted or when audit does not find them in their old address. These cartridges are marked absent, preserving all information about them, except for the scratch status. When these cartridges are later found by audit, they are reactivated.

- Absent cartridge support is enabled when the [“Variable: ABSENT\\_VOLUME\\_RETENTION\\_PERIOD” on page 126](#) is **not** zero. The default value is 5 days.
- Use `acsss_config` (option 3). For procedures, refer to the [“Variable: ABSENT\\_VOLUME\\_RETENTION\\_PERIOD” on page 126](#) under Option 3 - Set General Product Behavior Variables.

3. Update your configuration by one of the following methods.

- Dynamically, if PUT0501 to ACSLS 7.1 (or later) maintenance is installed:

**config acs *acs\_id***

- Statically, if Dynamic Configuration is not installed:
  - a. Shutdown ACSLS: **kill.acsss**
  - b. Update the ACSLS configuration: **acsss\_config**
  - c. Bring up ACSLS: **rc.acsss**

ACSLs cannot be running when you run `acsss_config`.

For more information, refer to [“Configuring or Updating your Library Hardware” on page 130](#) or under the [“Dynamic Configuration \(config\) utilities” on page 245](#).

## Adding New SL8500s

When additional SL8500s are added to an existing SL8500 library complex, the new ACSLS configuration must be updated. If the addition of new SL8500s causes the LSMs in the existing SL8500s to be renumbered, the cartridge addresses in those LSMs must be updated.

The cartridge addresses should be updated without losing the other information associated with the cartridges. This includes cartridge information, such as scratch status, pool, locks, and ownership, as well as cleaning cartridge usage counts, and the date and time cartridges were entered.

LSMs in connected SL8500s are numbered from top to bottom and right to left (as viewed from the CAP end).

**Figure 53. Four connected SL8500 Libraries**

Left		↔		↔		Right
Library 4		Library 3		Library 2		Library 1
LSM 12	P T P	LSM 8	P T P	LSM 4	P T P	LSM 0
LSM 13		LSM 9		LSM 5		LSM 1
LSM 14		LSM 10		LSM 6		LSM 2
LSM 15		LSM 11		LSM 7		LSM 3

### Adding a New SL8500 to the Left

In the case of the non-disruptive addition of a new SL8500 on the left (from the CAP end) of the original SL8500(s), you can use the host software (ACSL) to dynamically configure the additional drives and libraries. You do not need to re-cycle ACSLS if Dynamic Configuration is installed. Mount requests continue as normal to the existing SL8500s during this phase.

If cartridges were placed in cells in the new SL8500, an ACSLS audit must be run on the LSMs of the new SL8500 to add these cartridges to the ACSLS database.

The LSMs in the pre-existing SL8500s can be online during the audit.

#### ***Dynamically Configuring the New ACSLS Configuration***

1. Add new SL8500(s) to the library complex.
2. Backup ACSLS before you make the configuration change.
3. Update the ACSLS configuration dynamically (available if PUT0501 is installed):

```
config acs acs_id
```

You can also update the ACSLS configuration when ACSLS is down:

```
acsss_config
```

4. Backup ACSLS after you make the configuration change.

**Note:** If there are cartridges in the new SL8500, audit the LSMs (rails) in the new SL8500 to add these cartridges to the ACSLS database.

## Adding a New SL8500 to the Right

If you wish to add a new SL8500 to the right, the existing LSMs will be renumbered, as shown in the following figures

### **Considerations When Adding an SL8500 to the Right**

When new SL8500s are added to the right, all existing LSMs will be renumbered and cartridge addresses will change. The change to LSM numbers causes all existing cartridge addresses to change. If ACSLS attempts to mount a cartridge whose address changed, the mount will fail because ACSLS cannot find the cartridge.

- Quiesce mount activity until all cartridge addresses have been updated.
- Vary all LSMs in the ACS to diagnostic state to prevent mounts.
- Update the cartridge addresses by auditing the LSMs in the existing and new SL8500s in a specific sequence.

### **Dynamically Configuring the New ACSLS Configuration**

To update the ACSLS Configuration Dynamically after adding a new SL8500(s):

1. Vary existing LSMs in diagnostic state (`vary lsm_id dia`).

**WARNING:** These LSMs must remain in diagnostic state until they have been audited. Otherwise these problems will occur:

- Mounts will fail because cartridges cannot be found in their last-known addresses.
- Until the audits update the ACSLS map of the empty cells in the renumbered LSMs:
  - Enters of new cartridges will collide with existing cartridges.
  - Movements of cartridges to existing (renumbered) LSMs will collide with cartridges already in the cells,

2. Add the new SL8500(s) to the library complex.
3. Backup ACSLS (before making any configuration changes).
4. Update the ACSLS configuration dynamically using:

```
config acs acs_id
```

**Note:** As an alternative, you can update the ACSLS configurations while ACSLS is down. To do this:

- a. Bring ACSLS down using the `kill.acsss` command.
- b. Change the configuration with `acsss_config`.

- c. Bring ACSLS back up with `rc.acsss`.

**Note:** New LSMs are added when in an online state. Vary these LSMs to a diagnostic state until they have been audited.

5. Backup ACSLS (after the configuration change).
6. Audit the library to update cartridge addresses (since they were renumbered).

To do this, use the following the following sequence:

- a. First, audit each LSM in the pre-existing SL8500s that were renumbered.
  - **Audit the pre-existing SL8500s in a careful sequence to avoid losing cartridges. Audit each LSM (rail) in sequence from the highest numbered LSM to the lowest numbered LSM ID.** The audit will find all of the cartridges in their (renumbered) LSM addresses.
  - **For ACSLS, you must audit each LSM separately, and allow the audit of one LSM to finish before auditing the next LSM.** Audit will check the cartridges' former home cell addresses (in the LSM to the right), and then update the cartridges' addresses. The audits for the LSMs in the left most SL8500 will take a long time. The audits in all of the other SL8500s will be faster.

The reason the audits of the left most LSMs take longer is that they will trigger a cascade of Cartridge Recovery requests. When audit finds a cartridge that has a different home cell address, audit checks the cell address recorded in the database. If that cell has a different cartridge, Cartridge Recovery investigates that cartridge, etc. Because all of the LSM addresses have changed, these recoveries will cascade across the SL8500 library complex.
  - Keep the LSMs in diagnostic state until they have been audited. After they have been audited, you can vary them online. Audit has updated the addresses of cartridges in this LSM, so you can resume automated mount activity with the audited LSM's cartridges.
- b. Last, audit the LSMs in the newly added SL8500 (that have the lowest LSM IDs).

After these LSMs have been audited, they can be varied online and cartridges in them can be mounted on tape drives, as shown in the following figures.

**Figure 54. Original Configuration Before an SL8500 is added to the Right**

SL8500	SL8500	SL8500
LSM 8	LSM 4	LSM 0
LSM 9	LSM 5	LSM 1
LSM 10	LSM 6	LSM 2
LSM 11	LSM 7	LSM 3

**Figure 55. Correct Audit Order (shown in bold) and New Configuration**

First Four Audits	Second Four Audits	Third Four Audits	Last Four Audits
Now LSM 12 <b>(Audit # 4)</b>	Now LSM 8 <b>(Audit # 8)</b>	Now LSM 4 <b>(Audit # 12)</b>	New LSM 0 <b>(Audit # 16)</b>
Now LSM 13 <b>(Audit # 3)</b>	Now LSM 9 <b>(Audit # 7)</b>	Now LSM 5 <b>(Audit # 11)</b>	New LSM 1 <b>(Audit # 15)</b>
Now LSM 14 <b>(Audit # 2)</b>	Now LSM 10 <b>(Audit # 6)</b>	Now LSM 6 <b>(Audit # 10)</b>	New LSM 2 <b>(Audit # 14)</b>
Now LSM 15 <b>(Audit # 1)</b>	Now LSM 11 <b>(Audit # 5)</b>	Now LSM 7 <b>(Audit # 9)</b>	New LSM 3 <b>(Audit # 13)</b>

7. Backup ACSLS after the audits have completed.

## Procedures for Merging ACSs

SL8500 PTP supports the merging of multiple, separate SL8500s into a single ACS. To minimize the loss of information about cartridges and operational downtime, you should follow these recommended procedures.

**Note:** When the ACSLS configuration is updated, global cartridge address changes **are not** made.

For ACSLS, two scenarios are described below. In the first, the ACSs to be merged are numbered from right to left (as you face them from the CAP side). In the second, they are numbered from left to right. It is assumed that the resulting ACS will use the lower/lowest ACS ID.

## Merging ACSs Numbered from Right to Left

In the following scenario, the ACSs to be merged are numbered from right to left (as you face them from the CAP side).

**Figure 56. Existing Configuration – Three Separate ACSs**

ACS 2		ACS 1		ACS 0	
LSM 2,0		LSM 1,0		LSM 0,0	
LSM 2,1		LSM 1,1		LSM 0,1	
LSM 2,2		LSM 1,2		LSM 0,2	
LSM 2,3		LSM 1,3		LSM 0,3	

**Figure 57. Desired Configuration – Single ACS**

ACS 0					
LSM 0,8		LSM 0,4		LSM 0,0	
LSM 0,9	<b>P</b>	LSM 0,5	<b>P</b>	LSM 0,1	
LSM 0,10	<b>T</b>	LSM 0,6	<b>T</b>	LSM 0,2	
LSM 0,11	<b>P</b>	LSM 0,7	<b>P</b>	LSM 0,3	

### ***Procedure for Merging ACSs Numbered from Right to Left***

1. Vary all ACSs, except the right most ACS being merged, offline.  
This prevents mounts and dismounts while cartridge addresses are being updated.
2. Stop ACSLS.
3. Backup ACSLS before making any configuration changes.
4. Update the ACSLS configuration (while ACSLS is down) using `acsss_config`
5. Bring up ACSLS.
6. Vary the new LSMs (that were added to the ACS) to a diagnostic state.  
These LSMs were added in while in online state.
7. Backup ACSLS after the configuration change.
8. Audit the LSMs that were added to the ACS

The order in which they are audited does not matter. The entire ACS or all LSMs can be audited at once.

The cartridges will be re-activated.

9. Backup ACSLS after the audits are complete.
10. Vary the new LSMs online and resume normal automated processing.

## Merging ACSs Numbered from Left to Right

In the following scenario the ACSs to be merged are numbered from left to right.

**Figure 58. Existing Configuration - Three Separate ACSs**

ACS 0	ACS 1	ACS 2
LSM 0,0	LSM 1,0	LSM 2,0
LSM 0,1	LSM 1,1	LSM 2,1
LSM 0,2	LSM 1,2	LSM 2,2
LSM 0,3	LSM 1,3	LSM 2,3

**Figure 59. Desired Configuration - Single ACS**

ACS 0		
LSM 0,8	P	LSM 0,4
LSM 0,9	T	LSM 0,5
LSM 0,10	P	LSM 0,6
LSM 0,11	P	LSM 0,7
		LSM 0,0
		LSM 0,1
		LSM 0,2
		LSM 0,3

### ***Procedure for Merging ACSs Numbered from Left to Right***

1. Vary all ACSs offline.
 

This prevents mounts and dismounts while cartridge addresses are being updated.
2. Stop ACSLS.
3. Backup ACSLS before making any configuration changes.
4. Update the ACSLS configuration (while ACSLS is down) using `acs_ss_config`.
5. Bring up ACSLS.
6. Vary the new LSMs (that were added to the ACS) to a diagnostic state.

(These LSMs were added in while in online state).

7. Backup ACSLS after the configuration change.
8. Vary the new LSMs that were added to the left to a diagnostic state.  
You do not want to perform mounts and dismounts until you have audited these LSMs.
9. Audit the newly configured LSMs.  
The order in which they are audited **does** matter. Audit in this order:
  - a. **First, audit the LSMs in the SL8500(s) that retains the same ACS number first.**
    - You want audit to find these cartridges in their new LSM addresses before you audit the LSMs that now are assigned to these cartridges' old addresses.
    - The LSM addresses will be updated for all cartridges in these SL8500(s).
    - **For ACSLS, you must audit each LSM separately, and allow the audit of one LSM to finish before auditing the next LSM.**
    - Keep the LSMs in diagnostic state until they have been audited.  
After they have been audited, you can vary them online. Audit has updated the cartridge's addresses, so you can resume automated mount activity with the audited LSM's cartridges.
  - b. Last, audit the SL8500s that were merged into the lowest ACS. The cartridges in these SL8500s will be re-activated.
    - All of these LSMs can be audited at the same time.
    - The order in which these LSMs are audited does not matter.
10. Backup ACSLS (after the audits are complete)
11. Vary all LSMs online and resume normal automated processing.

## Removing PTPs and Splitting ACSs

You may decide to remove the PTP mechanisms that connect two SL8500s and split a single ACS into two separate ACSs. This configuration change is the reverse of merging two ACSs into a single ACS.

It is much easier to add the new ACS with the SL8500(s) on the left hand side of the split. This assigns the higher numbered LSMs to the new ACS and avoids renumbering the LSMs that remain in the existing ACS.

### ***Splitting ACSs, with the new ACS Created from SL8500(s) on the Left - Possible Scenarios***

**Figure 60. Existing Configuration – One ACS**

ACS 0						
LSM 0,12	P T P	LSM 0,8	P T P	LSM 0,4	P T P	LSM 0,0
LSM 0,13		LSM 0,9		LSM 0,5		LSM 0,1
LSM 0,14		LSM 0,10		LSM 0,6		LSM 0,2
LSM 0,15		LSM 0,11		LSM 0,7		LSM 0,3

**Figure 61. Desired Configuration - Two ACSs**

ACS 1				ACS 0		
LSM 1,4	P T P	LSM 1,0		LSM 0,4	P T P	LSM 0,0
LSM 1,5		LSM 1,1		LSM 0,5		LSM 0,1
LSM 1,6		LSM 1,2		LSM 0,6		LSM 0,2
LSM 1,7		LSM 1,3		LSM 0,7		LSM 0,3

### **ACSLs Procedure for Splitting an ACS**

With Dynamic Configuration, ACSLS can add ACSs while it is running. However, Dynamic Configuration does not make global cartridge address changes.

1. Backup ACSLS before the configuration change.
2. Vary the LSMs that will be moved to the new ACS offline.

This prevents mounts and dismounts while cartridge addresses are being updated.

The LSMs that are remaining in the existing ACS can remain online.

Mounts and dismounts in these LSMs can continue.

3. Remove the four PTP mechanisms that connect the SL8500s that you want to separate.
4. Reconfigure the existing ACS once the pass-thru ports have been removed.

This will remove the LSMs that are moving to the new ACS.

```
config acs acs_id
```

5. Add the new ACS.

```
config acs acs_id new
```

Alternatively, you can update the ACSLS configurations when ACSLS is down:

```
acsss_config
```

New LSMs are added in an online state.

6. Vary these LSMs in a diagnostic state until they have been audited.
7. Vary the new ACS to diagnostic state.
8. Backup ACSLS after the configuration change.
9. Audit the LSMs in the new ACS.

You do not want to perform mounts and dismounts until you have audited these LSMs.

The order in which they are audited does not matter. They can all be audited at once.

The cartridges will be re-activated.

10. Backup ACSLS after the audits are complete.
11. Vary the new LSMs online and resume normal automated processing.

LSMs can be varied online as soon as the audit of that LSM completes.

### ***Adding the New ACS on the Right Hand side of the Split***

If the new ACS is added on the right hand of the split, all cartridge addresses will be re-mapped.

As this is not recommended, a detailed procedure is not provided.

However, some considerations are:

- When the LSMs in the right hand SL8500(s) are removed from the existing ACS, the cartridges in these LSMs will be marked absent (if absent cartridge retention is active). They will be reactivated when the new ACS is audited.
- The LSMs in pre-existing ACS must be audited one LSM at a time to update the addresses of the cartridges in these LSMs. Audit the highest numbered LSM first, followed by the next lower LSM. The audit of one LSM must finish before the next audit is started.

## ■ Updating ACSLS After Adding HandBots

When you go from four to eight HandBots in the library, you need to update the ACSLS library configuration to reflect their presence.

You can select one of the following ways to do this:

- Install PUT0601 so ACSLS can dynamically update the database handtable with new records for the HandBots. With this option the library always remains online. This is the preferred method.
  - PUT0601S (Solaris)
  - PUT0601A (AIX)

For installation instructions, refer to the documents associated with the tar files for this PUT on the Customer Resource Center (CRC) website.

- If you have not installed this PUT, you must update the ACSLS library configuration by:
  1. Shutting down ACSLS.
  2. Running `acsss_config`.

This updates the ACSLS configuration to reflect the presence of the eight HandBots.
  3. Starting ACSLS.

## ■ ACSLS Dual TCP/IP Support on SL8500

Dual TCP/IP is an option that can be purchased for the SL8500. It provides two TCP/IP connections to the SL8500. However, you can continue to use the SL8500 with only one of the two connections operational. For more information on this feature, refer to the *StreamLine SL8500 Modular Library System Dual TCP/IP Feature* document. The latest copy can be found on the Customer Resource Center (CRC).

The purpose of dual TCP/IP is to automatically recognize and avoid a failing communication path. Since this is automated, there is no need for you to manually switch from an inoperative connection.

In order to take full advantage of dual TCP/IP support on the SL8500, the routing tables on the ACSLS server must be managed using the “route” command. This forces a route to the defined network interfaces on the SL8500 which in essence, creates a one to one relationship between interfaces. For further information on the UNIX “route” command, please refer to the manpages on your ACSLS server.

### Requirements

- Please coordinate with both your system administrator and network administrator to understand your current network environment and to identify all necessary IP addresses in advance.
- Please coordinate with your system administrator to either configure your network interface or to validate that it is configured properly.
- Verify you have the minimum software level and have installed the following PUT.
  - ACSLS 7.1.0 for Solaris SPARC
    - PUT0602S
  - AIX 7.1.0
    - PUT0602A
  - ACSLS 7.1.1 for Solaris 10 x86
    - PUT0602X

### Configuration

It is recommended that ACSLS keep two connections to the library open since ACSLS uses all active connections. If one connection is inoperative ACSLS uses the remaining operative connection, while continuing to try to re-establish communication on the failing connection.

The preferred configuration for dual TCP/IP implementations would be two network interfaces on two separate subnets for the ACSLS server as shown in Scenario 1. This provides maximum throughput and minimum resource contention with regard to network communications while adding a second physical connection improving reliability.

To configure two TCP/IP connections to a single library use the `acsss_config` utility or Dynamic Configuration (`config`). You must enter the number (2) of connections there are to the library and the IP addresses of the network devices.

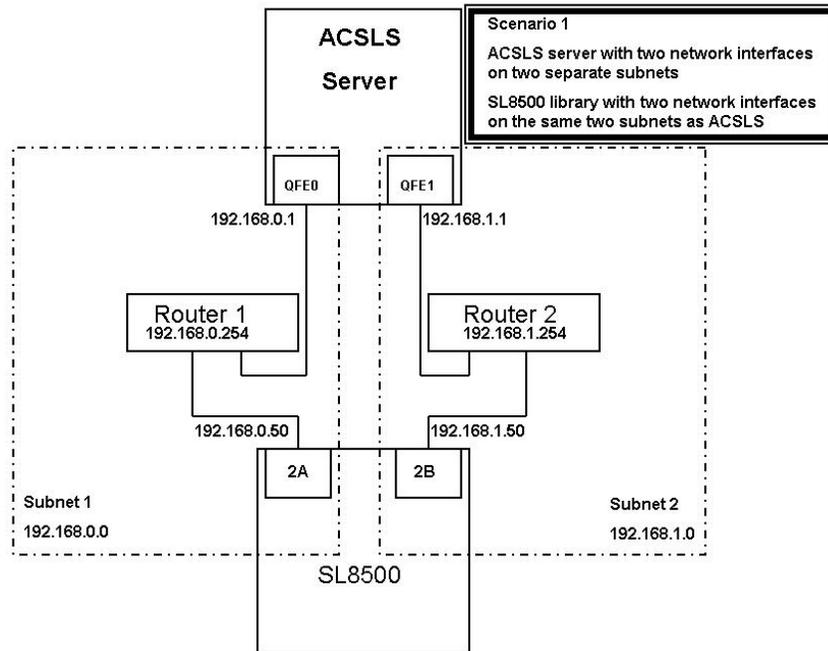
The following scenarios provide examples for configuring the ACSLS server. For instructions on configuring the SL8500 dual TCP/IP feature, please refer to the StreamLine SL8500 Modular Library System Dual TCP/IP Feature document.

These scenarios use private subnet IP addresses and will not be the same in your environment. These scenarios assume that your network devices have been configured and are functioning properly.

## Scenario 1

Scenario 1 is the preferred configuration for the dual TCP/IP feature.

In this configuration, the ACSLS server contains two network interfaces that reside on two separate subnets. The SL8500 also contains two network interfaces that reside on the same two subnets as the ACSLS server.



In this scenario, the SL8500 uses a one to one relationship with the network interfaces on the ACSLS server in which the:

- qfe0 interface on the ACSLS server only communicates with the 2A interface on the SL8500
- qfe1 interface on ACSLS only communicates with 2B on the SL8500.

Using the UNIX “route” commands, you force this relationship. To do this:

- For Solaris: as user `root`, type the following commands:

```
#route add 192.168.0.50 -ifp qfe0 192.168.0.254
```

```
#route add 192.168.1.50 -ifp qfe1 192.168.1.254
```

- For AIX: as user `root`, type the following commands:

```
#route add 192.168.0.50 -if qfe0 192.168.0.254
```

```
#route add 192.168.1.50 -if qfe1 192.168.1.254
```

The first route command routes any communication with 192.168.0.50 to go through qfe0 on the ACSLS server and then go through Router 1.

The second command routes any communication with 192.168.1.50 to go through qfe1 on the ACSLS server and then go through router 2.

You can validate that the routes are in the routing table by typing:

```
# netstat -r
```

```
Routing Table: IPv4
```

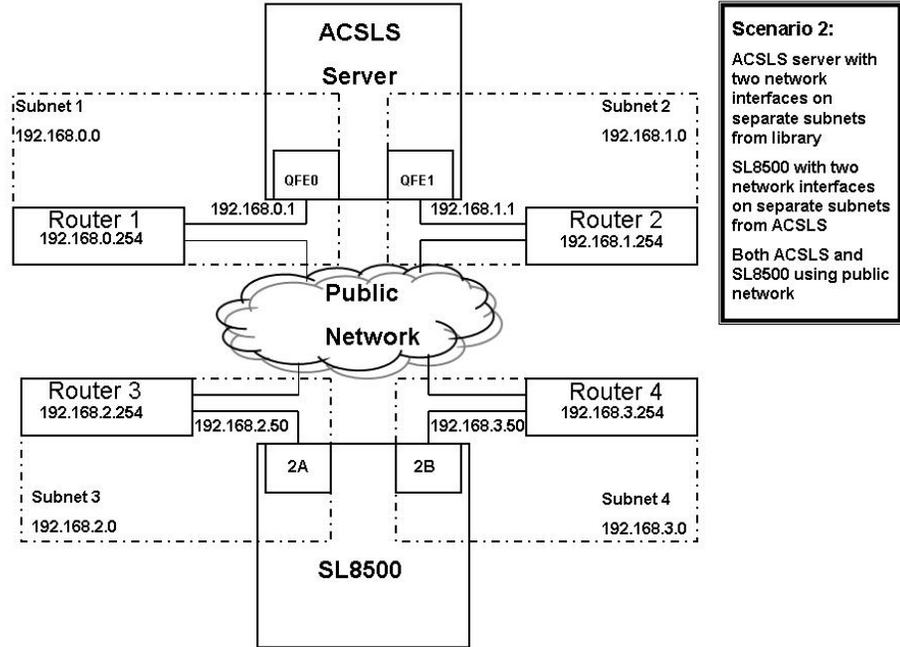
Destination Interface	Gateway	Flags	Ref	Use	
-----					
192.168.0.50	192.168.0.254	UGH	1	0	qfe0
192.168.1.50	192.168.1.254	UGH	1	0	qfe1
192.168.0.0	192.168.0.1	U	1	7	qfe0
192.168.1.0	192.168.1.1	U	1	0	qfe1
BASE-ADDRESS.MCAST.NET	192.168.0.1	U	1	0	qfe0
default	192.168.0.254	UG	1	33	
localhost	localhost	UH	4	77	lo0

You can see the first two entries are the ones that were just added. All communication with 192.168.0.50 will go through qfe0 and communication with 192.168.1.50 will go through qfe1.

**Remember:** You must also configure the SL8500 routing tables according to the instructions in the *StreamLine SL8500 Modular Library System Dual TCP/IP Feature* document.

## Scenario 2

Scenario 2 also shows two interfaces on the ACSLS server on two separate subnets but communication to the SL8500 will pass through the public network. Even with this difference, the commands stay the same.



**Scenario 2:**  
 ACSLS server with two network interfaces on separate subnets from library  
 SL8500 with two network interfaces on separate subnets from ACSLS  
 Both ACSLS and SL8500 using public network

Using the UNIX “route” commands, you force this relationship. To do this:

- For Solaris: as user root, type the following commands:  

```
#route add 192.168.0.50 -ifp qfe0 192.168.0.254
```

```
#route add 192.168.1.50 -ifp qfe1 192.168.1.254
```
- For AIX: as user root, type the following commands:  

```
#route add 192.168.0.50 -if qfe0 192.168.0.254
```

```
#route add 192.168.1.50 -if qfe1 192.168.1.254
```

The default routes for the ACSLS remain the same. The routes within the subnets will know how to route communication to the SL8500 through the public LAN and you are still forcing the one to one relationship with the interfaces. Again, this is seen using the following command:

```
# netstat -r
```

```
Routing Table: IPv4
```

```
Destination      Gateway          Flags Ref    Use
Interface
```

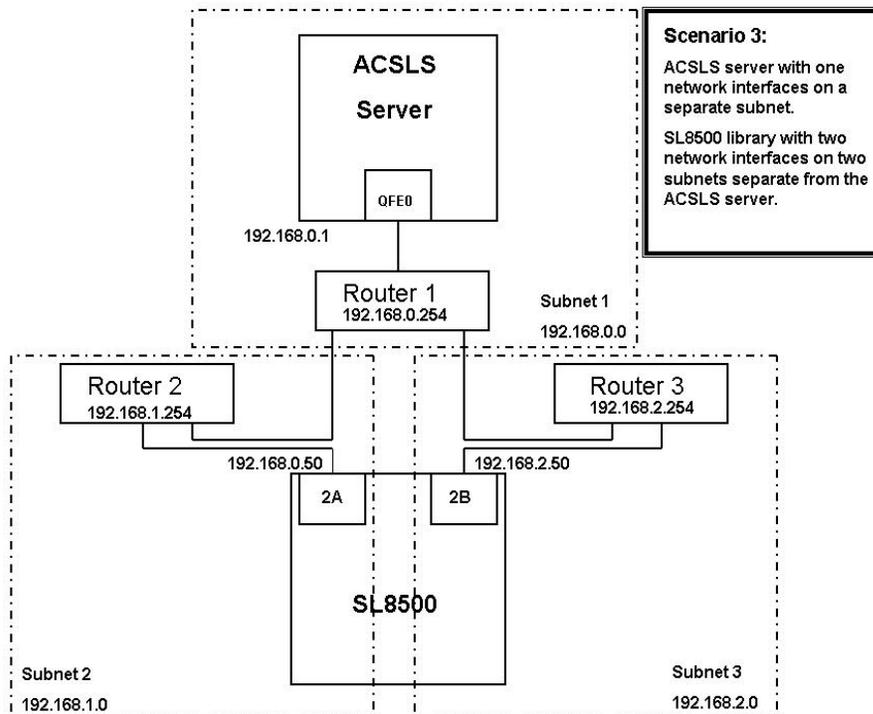
```
-----
```

192.168.0.50	192.168.0.254	UGH	1	0	qfe0
192.168.1.50	192.168.1.254	UGH	1	0	qfe1
192.168.0.0	192.168.0.1	U	1	7	qfe0
192.168.1.0	192.168.1.1	U	1	0	qfe1
BASE-ADDRESS.MCAST.NET	192.168.0.1	U	1	0	qfe0
default	192.168.0.254	UG	1	33	
localhost	localhost	UH	4	77	lo0

**Remember:** You must also configure the SL8500 routing tables according to the instructions in the *StreamLine SL8500 Modular Library System Dual TCP/IP Feature* document.

### Scenario 3

Scenario 3 is **not the preferred method** but will work. You can employ the dual TCP/IP feature on the SL8500 without using two network interfaces on the ACSLS server. This reduces reliability because if the ACSLS server network interface fails, communication with the library fails. If you do intend to implement this scenario, there is no need for special configuration on the ACSLS server. The default route is already established for normal network configuration, and it would be redundant to add specialized routes using the same default route. It is worth noting here that you still need to configure the SL8500 according to the instructions in the *StreamLine SL8500 Modular Library System Dual TCP/IP Feature* document.

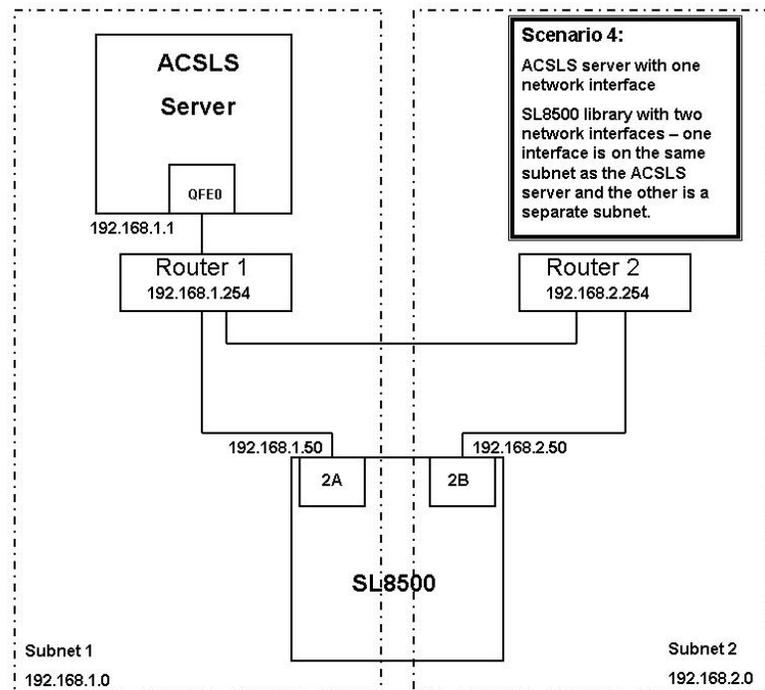


## Scenario 4

Scenario 4 also shows the ACSLS server using a single network interface while the SL8500 has two interfaces on two different subnets. This also **is not a preferred method** since there is no redundant interface on the ACSLS server.

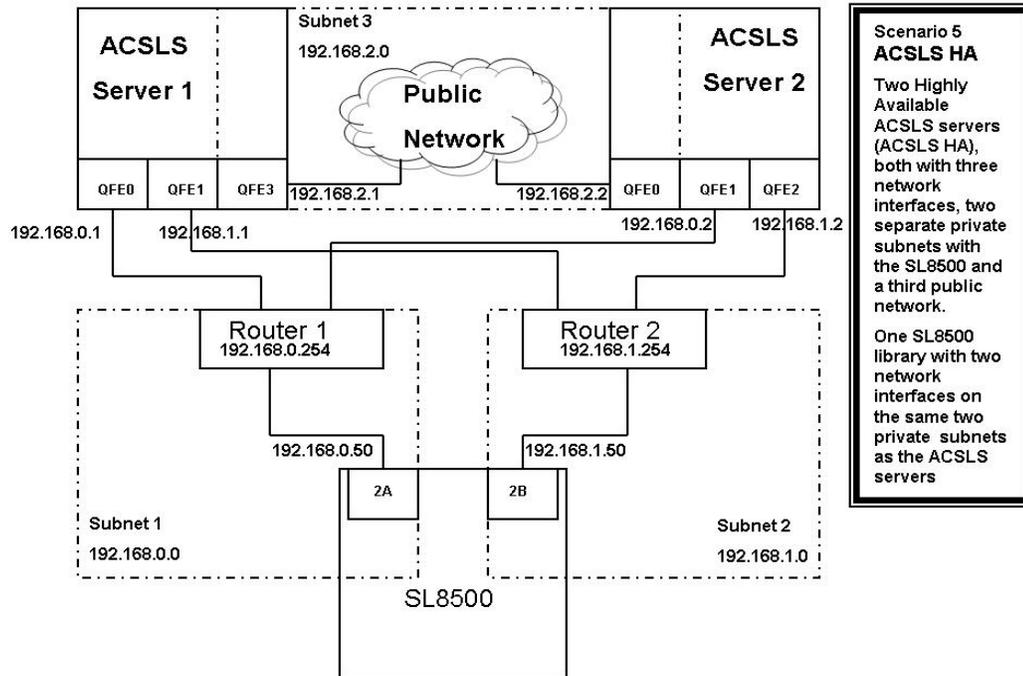
As in Scenario 3, there is no need for special configuration on the ACSLS server. Routing is already handled with a normal network configuration but the SL8500 still needs to be configured.

**Remember:** You must also configure the SL8500 routing tables according to the instructions in the *StreamLine SL8500 Modular Library System Dual TCP/IP Feature* document.



## Scenario 5

Scenario 5 shows Highly Available ACSLS (ACSLs HA) in a dual TCP/IP environment.



The big difference with this scenario is that ACSLS HA uses two different servers with each using different network interfaces. This means that custom route entries must be added to both ACSLS servers.

For the Solaris user:

- On ACSLS server 1, you would type:
 

```
# route add 192.168.0.50 -ifp qfe0 192.168.0.254
# route add 192.168.1.1 -ifp qfe1 192.168.0.254
```
- On ACSLS Server 2, you would type:
 

```
# route add 192.168.0.2 -ifp qfe1 192.168.0.254
# route add 192.168.1.2 -ifp qfe2 192.168.1.254
```

You must add the IP addresses for both servers to the SL8500 configuration. Please refer to the *StreamLine SL8500 Modular Library System Dual TCP/IP Feature* document.

It is important that you separate the SL8500 network interfaces over two different subnets when on ACSLS HA. The purpose of a Highly Available environment is to build in redundancy and eliminate single points of failure.

**Remember:** You must also configure the SL8500 routing tables according to the instructions in the StreamLine SL8500 Modular Library System Dual TCP/IP Feature document.

## Retaining Customized Routing Table Entries after a Reboot

Any customized routing table entries will be lost after a system re-boot. This is the nature of system routing tables and is expected behavior.

In order to support the Dual TCP/IP feature on the SL8500, it is necessary to add custom entries to the routing tables on the ACSLS server. When the ACSLS server is re-booted, all routing table entries are flushed and any necessary routes to the SL8500 are removed. Since this is the nature of the operating system, there are a couple of different ways to handle this situation:

1. You can create scripts to add custom routes to be initialized at boot time.

These scripts can then be placed in the rc directory structure for automatic execution at boot time. Please refer to your system documentation for details on the best way to implement this.

2. You can install the appropriate PTF for your operating system as discussed under Requirements. Please refer to the PTF README for installation instructions.

This allows the ACSLS startup scripts to add your custom routing entries at boot up time. The PTF includes new script entries that will check for a file that contains customized route table entries. Any entries found will be added to the routing table automatically using the UNIX route command.

For standalone ACSLS installations this is a desirable method to maintain route entries that are necessary for SL8500 support.

**Important:** This solution **will not work** if the ACSLS installation is a Highly Available ACSLS (ACSL S/HA) environment.

In this case, you need to use the first method to maintain routing tables.

ACSL S HA handles system initialization differently than a standalone ACSLS server. ACSLS HA relies on Veritas Cluster Server to manage its clustered resources which means ACSLS cannot be automatically started by way of the system RC mechanism at boot time. This is handled strictly by the Veritas agents, and the S87ACSL S startup scripts are never used. You need to add a script with the appropriate "route add" commands and locate it within the /etc/rc2.d directory structure. It is highly recommended that any customers with an ACSLS HA environment engage Sun Professional Services –

preferably the same PS consultant that originally installed the ACSLS HA system.

## Configuring ACSLS To Initialize Automatically At System Boot Time

To set up ACSLS to initialize automatically when the system is booted, execute **"fix\_rc.sh"** This interactive utility determines the current behavior of your system, reports that to you, and then asks you if you would like ACSLS to start automatically at boot time.

In the directory `$ACS_HOME/install`, as user **root**:

```
# ./fix_rc.sh
```

Your system is currently configured in such a manner that when it reboots, it will automatically restart the ACSLS.

Do you want your system configured so that when it reboots it automatically restarts the ACSLS? (**y or n**):

Simply answer **"y"** or **"n"** and the utility will complete the task.

Do you want your system configured so that when it reboots it automatically restarts the ACSLS? (y or n): **y**

```
fix_rc.sh 1864: Adding commands to automatically restart ACSLS on re-boot.
```

## Adding Custom Route To Be Initialized At Boot Time

To add custom routing entries:

1. cd to the following directory:

```
$ACS_HOME/data/external/ custom_routing.
```

This directory contains the template file **"custom\_routing\_tables.tpl"**

2. Copy this file and change the file name to "custom\_routing\_tables".

```
# cp custom_routing_tables.tpl custom_routing_tables
```

3. Edit (vi) the custom\_routing\_tables file and add your entries.

The file contains three fields.

- The IP address for the SL8500
- The name of the interface on the ACSLS server that you wish to establish the one-to-one relationship.
- The IP address of the default route for your subnet.

4. Follow the instructions in the "custom\_routing\_tables" comment section for the format.

**WARNING: Make sure that there are no blank lines.**

When your server reboots, ACSLS is automatically initialized, and your custom routes are added to the routing table.

5. Verify all routes in the routing table with:

```
# netstat -r
```

Please refer to your Unix man pages for complete documentation on both the "route" and "netstat" commands.

## Removing routing commands

Use the route command to remove any special routing commands that have been added erroneously or are no longer needed to the earlier configuration.

Example: As user `root`, type the following commands:

```
# route delete 192.168.0.50 192.168.0.254
```

This says to remove the route to 192.168.0.50 (the SL8500) using the default route of 192.168.0.254. The route is then removed.

# Library Partitioning

D

---

The SL8500 Modular Library System provides the ability to partition the library to support from one to four physical partitions. For more information on partitioning, refer to the *SL8500 Best Practices Guide*.

Partitioning the SL8500 library provides the following capabilities:

- ACSLS can manage open systems drives and cartridges, and HSC can manage mainframe drives and cartridges sharing the library
- More than one instance of ACSLS can access the library.
- An improvement in the protection or isolation of files.
- A higher level of data organization.
- An increase in user efficiency.

## ■ Software Requirements

ACSLS software requirements for partitioning the SL8500 library are:

- ACSLS 7.1/7.1.1 with PUT0701

If you use ACSLS HA, you also need:

- ACSLS HA 2.0 with PTF6514766

## ■ Guidelines for Partitioning

There are several steps that you need to follow when you are partitioning a library. They are:

1. Plan your new configuration.
2. Schedule an outage to partition or re-partition a library.

When a library is re-partitioned, the library drops its communication to all hosts managing all partitions, even if their partitions are not affected by the change.

3. Have your service representative license and enable partitioning on the SL8500.
4. Create your partitions using the SLConsole.

Refer to the SLConsole help for instructions.

5. Activate partitioning using ACSLS as described in [“Partitioning a Library or Changing Partition IDs” on page 491](#).

## Plan Your New Configuration

Plan your new partitioned configuration ahead of time.

- Organize the cartridges and drives for performance.

See the chapter on Library Content Management in the *SL8500 Best Practices Guide* and [“Improving Library Performance” on page 521](#).

- If you will do a lot of enters into and ejects from a partition, the partition should include at least one of the bottom three rails. Partitions with few enters and ejects can include the top rail.

## Minimize Disruptions when Partitioning an SL8500

When an SL8500 is partitioned or the partitions in an SL8500 are modified, access to your data is affected. Tape drives and cartridges that were managed by one host server, may be managed by a different host.

### Changes in Configuration

Changes in configuration occur when a:

- Library’s configuration has changed
- Library has been re-partitioned and the partition managed by ACSLS has changed (a message is displayed)
- Library capacity has changed and the cells reported as available to ACSLS have changed
- When LSM Configuration or capacity have changed

When the configuration may have changed, you are notified by ACSLS Event Log and Event Notification messages.

### Keeping Outages to a Minimum

The following procedures provided below ensure that outages are minimized and that you can continue to access your data. These procedures explain how to:

- Partition an un-partitioned SL8500 into two or more partitions.
- Modify the partitions in an SL8500 that is already partitioned.
- Merge a partitioned SL8500 so it is a single un-partitioned ACS.

## Challenges

The biggest challenges arise when LSMs (SL8500 rails) are removed from a partition.

- If you are combining multiple rails into fewer rails, you need to move the cartridges off the rails (LSMs) that will be removed from the partition to the LSMs that will remain in the partition. This applies whether you are partitioning a library that was not partitioned before or you are re-partitioning a partitioned library.
- Determine how many cartridges you have in the LSMs that you are emptying, and how many free cells are in the LSMs to which you are moving cartridges.

Use `free_cells.sh -a` to find out numbers of cartridges in these LSMs (allocated cells) and free cells.

- Shortly before you change the library configuration, move the cartridges away from the LSMs that will be removed from the partition.

Refer to the procedures for [“Moving Cartridges Before Removing an LSM from an ACS” on page 269](#) for details.

## ■ Partitioning a Library or Changing Partition IDs

These procedures describe how to partition an un-partitioned library or to change partition IDs on an existing partitioned library.

1. Move the cartridges before removing an LSM from an ACS.

If ACSLS manages one of the partitions in the new library configuration, you must first follow the procedures for [“Moving Cartridges Before Removing an LSM from an ACS” on page 269](#) to move the cartridges off of LSMs that will be assigned to another partition.

2. Quiesce the library before re-partitioning it and vary the ACS(s) and its ports offline.
3. Partition the library using the StreamLine Library Console™ (SLConsole).
4. The following then occurs after you press **<Apply>**. The library:
  - Rejects any new requests from ACSLS.
  - Drops all host connections.
  - Applies partitioning changes.
  - Accepts new host connections.
5. Reconfigure ACSLS to activate the partition changes.

Because the library was partitioned for the first time or it was re-partitioned and the partition ID changed, ACSLS cannot reconnect to the library. In order to reconnect, you must:

- a. Shutdown ACSLS using *kill.acsss*.
- b. Run *acsss\_config* to configure the new partition and select Option 8.

A message displays asking if the ACS is in a partitioned library. Refer to [“Configuring or Updating your Library Hardware” on page 130](#) for more information.

6. Start ACSLS with the new configuration.
7. Bring the ACSs and LSMs online.

**Note:** run *query lsm all* to view the list of LSMs

## ■ Re-partitioning a Library

These procedures describe how to re-partition an existing partitioned library.

1. Move the cartridges before removing an LSM from an ACS.

If ACSLS manages one of the partitions in the new library configuration, you must first follow the procedures for [“Moving Cartridges Before Removing an LSM from an ACS” on page 269](#) to move the cartridges off of LSMs that will be assigned to another partition.

2. Quiesce the library before re-partitioning it and vary the ACS(s) to diagnostic mode.

This causes ACSLS to reject new requests from clients while the ACS(s) remain available for dynamic re-configuration (after the re-partition).

3. Re-partition the library using the SLConsole.

The following then occurs after you press **<Apply>**. The library:

- Rejects any new requests from ACSLS.
- Drops all host connections.
- Applies partitioning changes.
- Accepts new host connections.

4. Vary the ACS(s) in all affected partitions to the diagnostic mode. Make sure all LSMs in these ACSs are also in diagnostic mode.
5. Reconfigure ACSLS to activate the partition changes.

**Note:** There may be no changes to a specific partition. The changes may only affect other partitions. In this case, no ACSLS configuration changes are needed.

**Note:** Because the library was not partitioned the first time or it was re-partitioned and the partition ID did not change, ACSLS can reconnect to the library.

To do this, select one of the following:

- Run Dynamic Config (*config acs acs\_id*) while ACSLS is running. This updates the ACSLS configuration to match the new library configuration.
- Shutdown ACSLS and run *acsss\_config* (option 8) to update the partition configuration in ACSLS.

6. Vary the ACSs and LSMs online.

**Note:** run *query lsm all* to view the list of LSMs

## ■ Changing a Partitioned ACS to an Un-partitioned ACS

1. Optionally, eject cartridges from the partition(s) managed by ACSLS.

If ACSLS will not be managing the non-partitioned library, you may want to eject cartridges from the partition(s) managed by ACSLS. This allows you to enter them into other ACS(s) that ACSLS manages.

You can use the *ejecting.sh* utility for ejecting multiple cartridges.

2. Quiesce the library before re-partitioning it and vary the ACS(s) and its ports offline.
3. Re-partition the library using the SLConsole.

The following then occurs after you press **<Apply>**. The library:

- Rejects any new requests from ACSLS.
- Drops all host connections.
- Applies partitioning changes.
- Accepts new host connections.

4. Reconfigure ACSLS to change the ACS to an un-partitioned library or to remove this ACS (partition) from its configuration.

Because the library was changed from partitioned to un-partitioned, ACSLS cannot reconnect to the library. In order to reconnect, you must:

- a. Shutdown ACSLS
- b. Run *acsss\_config* to configure the new partition.

Refer to [“Configuring or Updating your Library Hardware”](#) on page 130 for more information.

5. Start ACSLS with the new configuration.
6. Bring the ACSs and LSMs online.

## ■ CAP Behavior

### Sharing CAPs in a Partitioned Library

In a partitioned library, CAPs are a shared resource and need to be shared by all partitions. The process is as follows:

1. When a host uses a CAP to enter or eject cartridges, it reserves the CAP for its exclusive use.
2. The host must complete the enter or eject and leave the CAP empty and closed.

Should you need to terminate an enter on:

- ACSLS - the `cancel enter` command must be issued
- HSC - the “drain” command must be issued

3. The CAP is now available for use by any partition in the library.

### When Another Host Needs the CAP

If a host does not complete the enter or eject, the CAP may still contain cartridges for the affected partition. The CAP is still in a reserved status. This causes problems when other partitions need to use the CAP. To resolve this, the library must identify the host holding the reserve and terminate the enter or eject from that host.

In rare cases, you may not be able to access the host holding the reservation, either because the host crashed and cannot be restarted or you lack physical access or the security authority to issue the commands needed. In these cases, the CAP reservation can be overridden using the SLConsole.

**Overriding a CAP reservation using the SLConsole should be done only as a last resort.**

## Reserving the CAP

Consider the following when reserving the CAP:

- When a CAP is in use for an enter or eject operation, all 39 slots are reserved for that operation. The CAP cannot be subdivided.
- For addressing purposes, the CAP needs a location (ACS,LSM,CAP#). The LSM address is associated with the second rail in each library. For example: ACS#,1,0 for CAP A and ACS#,1,1 for CAP B
- Reserving the CAP

The CAP is a common component among partitions. Each host will:

- reserve the CAP to use it for an enter or eject
- release the CAP when the enter or eject is terminated. Another host can then use it.

## Terminating a CAP Reservation

1. When ACSLS finds that a CAP is reserved by another partition, it reports the partition ID and host ID holding the reservation.

You then need to contact the host owning the reserved CAP and request them to complete the enter or eject.

Once this is done, the empty CAP is then unreserved and available.

2. If the CAP reservation cannot be released by the host holding it, the SLConsole command is used to override the CAP reservation.
  - a. The SLC command displays a warning to the host informing it that the reservation from the specific partition owner will be removed.
  - b. The SLConsole queries the CAP to find the partition owner/requester/reserver.
  - c. The SLC displays the partition name (HLI1, HLI2, HLI3, HLI4, or Default) in the warning.
3. The library sends a CAP Overridden Message to the host that held the reservation.

This host should terminate the enter or eject task that was in progress.

4. The SL8500 now owns the CAP.

The CAP must be empty and closed before it is available to be used by another host. If the:

- CAP has cartridges in it, you must empty the CAP before a partition can use it.

## Library Partitioning

- CAP is open, you must close it. Then the library can audit it and determine if there are any cartridges in the CAP. Until the CAP is closed, it cannot be made available to any partitions.
  - CAP is empty, it will be locked, and it is available to be reserved by any partition.
5. The CAP is now unreserved and is available for use by any partition.

# ACSL S Support of the SL500



---

The SL500 library is a SCSI-attached, single-LSM library. It requires either a fibre channel or a LVD SCSI connection to the ACSLS server for control path operations. This chapter discusses ACSLS support for the SL500 library.

## ■ ACSLS Connectivity

The ACSLS server needs a Host-Bus Adapter (HBA) that is compatible with the interface card in the SL500 library (either LVD SCSI or Fibre). Refer to the ACSLS Product Information Guide for available feature codes and part numbers for library connections.

When you run `$ACS_HOME/install/install_scsi_sol.sh`, the system is probed for the existence of new hardware: either a JNI card or an LSI Logic card. If either of these HBA cards is present on the system, but the corresponding driver is not in place, the routine displays a prompt that asks you for permission to locate the appropriate driver package and install it.

## ■ ACSLS and SL500 Library Differences

### Library Configuration

- ACS, LSM - SL500 library is an ACS with a single LSM (1 ACS and 1 LSM)
- The SL500 does not currently support the pass-thru-port
- The SL500 has at least 1 base module, and can have up to 4 expansion modules.
- ACSLS considers each SL500 module as a panel.

### Identifying Library Locations

#### Addressing schemes

ACSL S addressing is ACS, LSM, panel, row, column

- SL500 addressing is LSM within the library, module, row, column
- ACSLS addresses are zero-based. They start with panel 0, row 0, and column 0.

- SL500 libraries are one-based. They start with module 1, row 1, and column 1.
- In order to ensure consistency between the SL500 library and ACSLS internal addresses, ACSLS defines placeholders for panel 0, row 0, and column 0.

## **Panels**

- Panel 0 = not installed
- Panel 1 = base module
- Panels 2-5 = expansion modules

## **Row numbers**

- Within each module (panel), rows are numbered 1-12
- Rows are numbered consecutively from the top down
- Numbering is identical between the SL500 and ACSLS

## **Column numbers**

- Numbered from 1 to 11 (maximum)
- The current SL500 configuration is limited to 9 columns
- Columns 1-4 are on the left side from front to back
- Columns 5-8 on are the right side from back to front
- Column 9 is the back wall where the drives are located
- Numbering is identical between the SL500 and ACSLS

## **Drive addresses**

SL500 addressing is - LSM, module, row, column

- Row is equal to the drive number or slot number
- Drive number is 1-2 on the base module and 1-4 on the expansion modules
- Column number is always 9 for drives
- ACSLS addressing is ACS, LSM, panel, drive
- Drive number is equal to a row or slot

**Figure 62. Example of drive addressing**

	ACSLs Drive Identifier	SL500 Drive Address
Base module	0, 0, 1, 1	0, 1, 1, 9
	0, 0, 1, 2	0, 1, 2, 9
	0, 0, 2, 1	0, 2, 1, 9
Drive expansion module	0, 0, 2, 2	0, 2, 2, 9
	0, 0, 2, 3	0, 2, 3, 9
	0, 0, 2, 4	0, 2, 4, 9
	0, 0, 3, 1	0, 3, 1, 9
Next expansion module	etc.	etc.

## ■ ACSLS Limitations

- Maximum number of SL500 libraries = 31
- Maximum number of LSMs = 127
- Maximum number of panels in a SL500 library = 5
  - Expansion modules add 1 panel each
- An SL500 with the base module only has just 1 panel
  - The “move” command cannot be used
- The SL500 firmware does not support unlabeled cartridges. This means that the “venter” command cannot be used

## ■ Configuring SL500 Library Settings

The SL500 supports new configuration settings that affect ACSLS. They are:

- CAP as storage – CAP cells
  - Only allowed in expansion modules
  - Base module CAP is always used as a CAP
  - Configured on a per-module basis
  - Any CAP used as storage cannot be used in enter/eject operations
- Reserved cells are for library use only (diagnostic, cleaning cartridges)
  - Only affects base module

- Number (n) can be configured from 0 to 9 (max row in base module)
- The first (n) cells in column 1 will be reserved
- These cells are considered inaccessible to ACSLS

## ■ SL500 CAP Behavior

The CAP in the base module has one five-slot magazine. When a drive expansion module is added, the CAP in the drive expansion module has two five-slot magazines. In the case of expansion modules, the CAP can be configured for 10 additional storage slots per expansion module. In addition:

- CAP cells, other than those on a base module, can either be configured as CAP cells or as storage cells.
- The base module CAP is always used as a CAP.
- The SL500 must have one active CAP for entering and ejecting cartridges.
- If there are multiple CAPs, the SL500 library always reports to ACSLS that it has one CAP.
- The doors to all of the CAP sections lock and unlock as one unless the CAP has been configured as a storage cell.

If a module has been configured to use CAP cells as storage cells, the CAP section is not affected by library operations that access the CAP.

- When ACSLS audits the CAP, it examines all cells.

## ■ Auditing the Library

A self-audit of the library occurs when you:

- Power-on or re-initialize the library
- Open and close the door

After the self-audit is complete you need to update your database using ACSLS. The following discusses how to audit the database after you have:

- Added one or more modules, removed one or more modules, or swapped modules.
- Manually added or removed cartridges through the door
- After library configuration settings have changed

## After adding, removing, or swapping modules

After a SL500 module has been added, removed, swapped, or if you are auditing the library for the first time, perform the following steps:

1. Vary the SL500 library offline to ACSLS and power down the library.
2. Perform the library change, such as adding a module.
3. Power on or re-initialize the SL500 library.
4. Terminate ACSLS (if it is running).

```
idle
kill.acsss
```

5. Run `acsss_config`.
6. Start up ACSLS.

```
rc.acsss
```

7. Perform an audit of the SL500 panel(s) affected.

## After manually adding or removing cartridges through the door

If you have manually added or removed cartridges from the LSM through the library door you must synchronize the databases. To do this, perform an audit as described under [“audit” on page 294](#).

## After entering or ejecting cartridges through the CAP

If you have entered or ejected cartridges from the LSM through the CAP using the ACSLS `cmd_proc`, the database is automatically updated.

## After changing library configuration settings

If you have changed library configuration settings you should reconfigure ACSLS using `acsss_config` and then audit the library. You should also reconfigure and audit the library after changing the following library settings:

- Number of reserved cells
- CAPS as storage cells

To do this:

1. Vary the SL500 library offline to ACSLS and power down the library.
2. Perform the library change.

3. Re-initialize the SL500 library.
4. Terminate ACSLS (if it is running).

```
idle  
kill.acsss
```

5. Run `acsss_config`.
6. Start up ACSLS.

```
rc.acsss
```

7. Perform an audit of the SL500 panel(s) affected.

# Troubleshooting

## F

This chapter describes how to troubleshoot library and ACSLS errors. You can resolve some errors, but others may require assistance from StorageTek. This chapter describes the following troubleshooting topics:

- ACSLS and library hardware error recovery
- ACSLS event log
- Logs of ACSLS Installation and Utilities
- Troubleshooting Library Connections
- Verifying the SCSI Library Connection
- Gathering ACSLS Diagnostic Information

## ■ ACSLS and Library Hardware Error Recovery

This section describes error recovery that ACSLS and the library hardware provide. If an individual process or a non-critical library component fails, ACSLS records the error in the ACSLS event log and continues to provide library services with the unaffected parts of the system.

If a major system failure occurs, however, library operations are suspended until the error is corrected. The following sections describe how ACSLS and the library hardware respond to communications, hardware, and software failures.

Use the ACSLS event log to gather information about ACSLS and library hardware errors (see [“ACSLS Event Log” on page 505](#)).

## Communications Failures

Communications failures include the failure of communications lines between ACSLS and an LMU or between an LMU and an LCU. Either hardware or software errors can cause these communications failures.

Communications software failures also include the failure of interprocess communication between ACSLS and the CSI (client interface) or `cmd_proc`.

If ACSLS cannot communicate with another library component, it logs an error and retries until contact is established or until a system-defined timeout period is reached.

## Hardware Failures

Hardware reliability and redundancy can allow library operations to continue even if one component fails. For example:

- A dual-LMU configuration switches to the standby LMU if the master fails; for more information, see [“Managing a Dual-LMU Configuration” on page 175](#).
- A dual-LAN client configuration switches to the backup LAN if the primary fails; for more information see [“Managing a Dual-LAN Client Configuration” on page 177](#).

Other hardware failures, however, can suspend library operations until the failed hardware is repaired or replaced. The following list describes typical hardware failures and their effect on library operations:

### **LSM Robot Failure**

Complete loss of robot function makes the affected LSM unavailable.

### **Loss of Robot Hands, Cameras, or Lights**

Library processing in an LSM can continue in a degraded mode if the robot loses only one hand, camera, or light. If both hands, cameras, or lights fail, however, the LSM becomes unavailable.

**Hint:** If your LSM fails and goes off line but the data path to the tape drives is still operational, you can still manually load cartridges into the library drives.

### **CAP Failure**

If a CAP fails, you cannot enter and eject cartridges directly into the affected LSM through that CAP. All other library processes can continue normally. If the affected LSM has multiple CAPs, you can use another CAP. If the affected LSM is connected to another LSM via a PTP, you can use the second LSM's CAP for enter and eject operations.

## Software Failures

Major software failures include a system crash, a database failure, or a library configuration inconsistency. These errors result in loss of library operations in all affected ACSs. After the problem is corrected, ACSLS goes through automatic recovery procedures to restore library operations.

## Tracking Software Problems

ACSLs and the Operating System provide the following software facilities:

### ACSLs Event Log

This log contains a time-stamped history of significant events. Use it to help troubleshoot software problems. See [“ACSLs Event Log” on page 505](#) for a detailed description of this log.

## Applying Software Maintenance

Please check the Customer Resource Center (CRC) on a regular basis for fixes and enhancements to supported ACSLS releases. The CRC also has the most current versions of the ACSLS documentation.

You must login to the CRC to view product support information.

## ■ ACSLS Event Log

The ACSLS event log contains information about library events and errors. All ACSLS components log events to the event log through the centralized event logger. The base event log, which is automatically created when ACSLS is installed, is contained in the file `$ACS_HOME/log/acsss_event.log` which is usually `/export/home/ACSSS/log/acsss_event.log`

Logged events include the following:

### Significant Events

These are normal events that can help you manage the library. For example, events are logged when an audit is initiated or terminated, a device changes state, or a CAP is opened or closed.

### Library Errors

Both fatal and nonfatal hardware and software errors are logged. Examples include LSM failures, problems with cartridges, database errors, interprocess and library communications failures, and software failures not normally handled by the operating system.

## Using the Event Log

You should browse the event log periodically to help manage ACSLS and the library. Event log entries are particularly useful after:

- An audit
- A hardware or software failure
- ACSLS recovery

See *“ACSLs Messages”* for descriptions of event log messages.

When you log into ACSLS as the `acssa` user, a window with a tail of the event log is included on the standard terminal display; for more information, see [“The `acssa` and `acsss` User IDs” on page 2](#).

**Hint:** To tail the event log from another login, enter:

```
acs_tail $ACS_HOME/acsss_event.log
```

You can view the event log in a text editor, for example:

```
view acsss_event.log
```

## Managing the Event Log

You use the `acsss_config` configuration program to specify the following:

- Event log size and number of rollover files
- Pathname of the directory that contains the event log
- Event log date/time format
- Whether the event log logs volume enters in automatic enter mode
- Whether the event log logs database volume additions and deletions

For more information, see [“Setting Event Logging Variables” on page 120](#).

The following are errors that may occur as part of event log processing.

- If a communication failure occurs while the event logger is sending a message to `cmd_proc`, the unsolicited message is lost.
- The following unsolicited message is displayed if the event logger is unable to access or write to the event log file. This may be due to incorrect permissions on the directory or the file.

```
Event log access failed
```

## Using `greplog` to Search Even Logs

`greplog` does intelligent ACSLS Event Log searches. When `greplog` finds a match, it returns all lines of the Event Log message, not just the line containing the string that was matched. This lets you see the message's date and time stamp, message number and the function text. Because a `greplog` search provides the full context of the message, it facilitates research into ACSLS events.

### Format

The format for searching the Event Log using `greplog` is as follows:

```
greplog [-iv] pattern file_1 file_2 ... file_n
```

## Options

- i  
tells `greplog` to ignore the case of the search pattern expression.
- v  
specifies that `greplog` displays all of the entries in the log file except those entries which match the pattern expression.
- pattern:  
specifies the search pattern expression
- file\_1 file\_2 ... file\_n  
`greplog` accepts multiple file parameters and wild card expressions in the file list.

## Usage

`greplog` can be used to search for a pattern within the `acsss_event.log`. For each hit, it returns the entire Event Log entry containing the search pattern expression, not simply the single line containing the word. For example, a typical message contains the following information:

```
mm-dd-yy hh:mm:ss component_name [nn]:
message_number classification mod_id mod_ver mod_line
function: One or more lines of text
```

After retrieving matching messages with `greplog`, use the date and time in the message(s) to view that time period in the full event log and see the context for the event.

## Examples

To search the Event Log for all occurrences of message number 1392:

```
greplog 1392 acsss_event.log
```

To search the Event Log for all messages about volume CART89:

```
greplog CART89 acsss_event.log
```

To search all archived copies of the Event Log for messages about tape mounts:

```
greplog -i mount event*.log
```

## Note

The archived copies of the Event Log are named `event1.log`, `event2.log`, etc. `event*.log` searches all of them.

## ■ Logs of ACSLS Installation and Utilities

The `$ACS_HOME/log (/export/home/ACSSS/log)` directory, which contains the ACSLS event log, also contains logs from the ACSLS installation and for each of the ACSLS utilities. The contents of these logs are summarized in the entries for `/export/home/ACSSS/log` in [Table 2 on page 13](#).

Entries are appended to these logs whenever you run the corresponding utility or script.

**Hint:** The utility logs do not have the same file sizing and rollover options that you can use to automatically manage and ACSLS event log. However, `monitor.sh` in `$ACS_HOME/diag/bin` can be used to monitor and automatically archive any log. For information on `monitor.sh`, see `README.txt` in `$ACS_HOME/diag/bin`.

The `acsss_config.log` is a report of library hardware configured by the `acsss_config` program. StorageTek recommends that you retain this log, which provides a useful record of your library hardware.

The `bdb_event.log`, `export_event.log`, `import_event.log`, `install.log`, `rdb_event.log`, `sd_event.log`, and `volrpt.log` record success and failure entries each time you run the corresponding script or utility. These logs are most useful, therefore, right after you run the utility. Especially if the utility failed, review its log, correct any errors, and rerun the utility until it runs successfully. After the utility runs successfully, you may want to delete its log to free disk space and to ensure that when you next run the utility, the log is recreated and contains only current entries. If you cannot successfully run the utility, however, save the log, which can help StorageTek resolve the problem.

The `cron_event.log` records events for the cron job that periodically runs the `full_disk.sh` script. The `full_disk.sh.log` records events for the `full_disk.sh` script, which does the following:

- Logs warning messages in the ACSLS event log if the ACSLS home directory (`$ACS_HOME/acsss`, usually `/export/home/ACSSS`) exceeds 85% full
- Backs up the ACSLS database as described in [“Automatic Database Backup” on page 215](#).

Because the `cron_event.log` and the `full_disk.sh.log` record periodic events, you should not remove these logs. To conserve disk space, however, you may want to periodically edit these files to remove older, less meaningful entries.

## ■ Troubleshooting Library Connections

To troubleshoot Library Connections use the ACSLS tools appropriate for your platform.

StorageTek has provided a set of tools to verify the connection between the ACSLS server and the attached LMU or LSMs. This section is laid out in two parts. The first part describes various ACSLS and OS diagnostic utilities that can be used to test and verify hardware connections to the library. The second part offers a troubleshooting strategy to use in the event of problems with library communication.

### Part 1: ACSLS and OS Diagnostic Utilities

#### ck\_port\_node.sh

This utility is valid only for Solaris systems. It extracts a long listing of the port node for all defined ports in ACSLS. It reveals port permissions, port ownership and the hardware device path to the port. The hardware path reveals the communication path from the most internal bus to the unique controller, to the most external connection on the box itself. All ACSLS ports are character devices that are owned by root and available to all users (rw-rw-rw-).

#### *Example*

```
$ ck_port_node.sh
crw-rw-rw- 1 root sys 165, 1 May 31 15:35
/dev/./devices/pci@1f,0/pci@1/scsi@3/mchanger@6,0:
character

crw-rw-rw- 1 root sys 20, 1 Jun 7 15:35
/dev/./devices/pci@1f,0/pci@1,1/ebus@1/se@14,400000:b
```

In the case of the mchanger device, the target ID of the library is the number 6 in mchanger@6.

In the case of the serial device, the specific port is the last character in the listing (port b).

#### probeFibre.sh

This utility makes use of the fibre HBA software to probe any fibre-attached library devices. It must be run as `root`. It is a method you can use independently of the `mchanger` driver to verify whether a fibre-attached library is connected and responding. If fibre-attached libraries are unable to respond to this command, it is a sign that the library, the library connection, the fibre HBA, or the HBA driver is faulty. A good response from `probeFibre.sh` indicates that conditions are in place to enable the `mchanger` device for each responding library. For each library that responds, the utility will display the

vendor, model, code revision, and world-wide port name of the attached library.

## probescsi.sh

This utility will send a SCSI inquiry test packet to a SCSI-attached LSM and verify the response. If the communication is good, the LSM will send a return packet revealing the make, model number and microcode level of the LSM. Since it communicates directly with the LSM, this routine cannot be used while ACSLS is running.

You must pass the appropriate port name as an argument to `probescsi.sh`.

### **Example**

```
$ probescsi.sh /dev/mchanger0
/dev/mchanger0 is communicating to an LSM: STK L180 level
0221
```

A positive response verifies the device driver and the essential hardware path to each library.

## showDevs.sh

This utility will probe each mchanger device that is found in the `/dev` directory. If the device responds, the utility will display the details that were reported by the device, including the adapter, the device name, node name, world-wide port name, target, LUN, vendor id, model number, code revision, serial number, and geometry including the number of cells, and the number of drives. Since the actual display will depend on the specific library attached and the permissions of the user, only a subset of this information may be included in any specific display. This utility may be run as user `acsss` or as `root`.

## testports

This utility tests all of the library connections that have been defined in the ACSLS database. Port connections are defined using the `acsss_config` utility. See [“Configuring or Updating your Library Hardware” on page 130](#). If you have not yet defined library communications with `acsss_config`, you must use `testlmu` or `probe-scsi.sh` (see below) to test each individual library connection. Since `testports` actually communicates through the same port defined in ACSLS, the `testports` utility will not work while ACSLS is running.

For each library that is defined, `testports` will send a test packet to the library and then wait for an appropriate response from the library. In the response, the library reveals either its model number and microcode level or its compatibility code level. If the connection is good, `testports` will display the port name that is used in the connection and the model and/or code confirmation that was returned from the library.

**Example**

```
$ testports
/dev/mchanger0 is communicating to an LSM: STK L180 level
0221

/dev/ttyb is communicating to an LMU at compat level 13
```

A positive response to `testports` verifies the device driver and the essential hardware path to each library.

**testlmu**

This utility will send a test packet from the ACSLS server to a serial-attached or a network-attached LMU and will verify the response. If the connection is good, the LMU will respond with its code compatibility level. Since `testlmu` uses the ACSLS port to communicate to the LMU, the port must be offline to ACSLS. If ACSLS is holding the port online, you will get a false indication of failure with `testlmu`.

You must pass the appropriate port name as an argument to `testlmu`.

**Example**

```
$ testlmu /dev/ttyb
LMU Responded, compat level 12
```

A positive response verifies the device node and the essential hardware path to the library.

**System Utilities: AIX****lsdev**

On AIX systems, you can identify the hardware device path using `lsdev`. The hardware path reveals the communication path from the internal PCI bus to the most external connection on the system.

**Example**

```
$ lsdev -C -c tty
tty0 Available 01-C0-00-00 Asynchronous Terminal
```

In this example, the unique port identifier was "C0"

**Example (SCSI)**

```
$ lsdev -C -s scsi
mchanger0 Available 04-05-00-6,0 Other SCSI Tape Drive
```

In this example, the unique SCSI controller in this path was identified as "05". The SCSI device in this example resides at target-6, logical unit-0.

## diag

This utility bypasses all software drivers and sends a very simple inquiry command directly from AIX to the library. A valid response will reveal that the hardware connection is good and that the library can communicate.

To use `diag` in order to verify the hardware path to the library here is the process:

1. Become `root` user with:  
`login` or `su`.
2. Enter the command:  
`diag`
3. At the first screen, which is informational, press `<Enter>`.
4. At the `Function Selection` screen, scroll down until the “`Task Selection()`” option is highlighted.  
 Press `<Enter>`.
5. In the `Task Selection List`, scroll down until `SCSI Bus Analyzer` is highlighted.  
 Press `<Enter>`.
6. In the `Resource Selection List`, identify which `SCSI I/O controller` is used for your `STK library`.  
 If you are not sure which controller to specify, use `lsdev -C -s scsi` as described above.
7. Highlight the appropriate controller.  
 Press `<F7>` to commit that device for this test.
8. The first `SCSI Bus Exerciser` screen is informational. Press `<Enter>`.
9. In the second `SCSI Bus Exerciser` screen, press the `[[Tab]]` key to select the target address that is assigned to the `STK library`. If you are not sure which target to specify, consult the configuration option in the operator panel on your library.  
 Alternatively, you can use the AIX command `lsdev -C -s scsi` as described above.
10. When the correct target ID has been selected, press `<F7>` to commit the test.
11. In the third `SCSI Bus Exerciser` screen, the system will ask you to confirm the test to send a `SCSI inquiry` command across the bus to the library.

Press <Enter> to confirm the test.

- The library should respond with its make and model. The ASCII character response will be displayed in its hexadecimal form.

To determine whether the test was successful, look for the character string "53 54 4B" (STK).

### **Example**

The following Inquiry data was returned:

```
08 80 02 02 33 00 00 00      53 54 4B 20 20 20 20 20
4C 31 38 30 20 20 20 20      20 20 20 20 20 20 20 20
31 31 30 30 20 20 20 20      20 20 20 20 20 20 20 20
20 20 20 20 20 20 20 20      00 00 00 00 00 00 00 00
```

The ASCII characters (in boldface) translate "STK L180"

## **System Utilities: Solaris**

### **probe-scsi-all**

This utility actually runs beneath Solaris at the PROM monitor level. This low-level utility bypasses all software drivers and sends a simple 'inquiry' command directly to the library. A valid response will display the make, model and code level of the library. This reveals whether the hardware connection is good and the library can communicate.

On Sun Ultra systems, it is necessary to abort UNIX and also reset the system before running probe-scsi-all. Use the following procedure:

- Become root user with:

```
login or su.
```

- Halt the system with:

```
init 0
```

- At the ok prompt, enter the command:

```
reset-all
```

The screen will go blank for a few seconds.

- When the system begins to boot, interrupt the boot sequence with:

```
Stop-A (Press the Stop key together with the letter A.)
```

- At the ok prompt, enter:

```
probe-scsi-all
```

**Example**

```
ok probe-scsi-all
This command may hang the system if Stop-A or the halt
command has been executed.
Please type reset-all to reset the system before executing
this command.
Do you wish to continue? (y/n) y
/pci@1f/pci@1/scsi@3,1

/pci@1f/pci@1/scsi@3
Target 6 Unit 0 Removable Device type 8STK L180 0221
```

In this example, the system found two SCSI controllers. The first controller, SCSI@3,1, had nothing connected. The second one, SCSI@3, was connected to a device at target-6. That device was a StorageTek L-180 library running microcode version 0221.

## Part 2: Strategies for Troubleshooting Host-Library Communication Problems

There are numerous possibilities that might account for communication problems between ACSLS and the library. The root cause may be related to hardware problems, software problems, driver problems, or device node settings. This section is intended to help you isolate the various possibilities down to one or more causes.

### Troubleshooting a Serial Connection

To verify the serial hardware connection, you can use either of two utilities. If the serial port has already been defined in ACSLS then you can use the command `testports`. Otherwise, use `testlmu`, passing the desired `port_name` as an argument.

If the test response is positive, then you have verified a good hardware connection. If the response is negative, then you should consider each of these possibilities:

1. Verify a valid port definition.

Make sure that the port you specified (or the port that was defined in `acsss_config`) is the correct port name. You should identify the physical connector on the machine that maps to that particular port address, and then make sure that the LMU cable is connected to that specific port.

2. Verify a valid port node.

On a Solaris machine, run the ACSLS utility `ck_port_node.sh`. If the port name has not been defined in `acsss_config` then you will need to pass the port name as an argument. This utility will return a listing of the

low-level port node for the specified port. From the listing, you should verify the following:

- a. The leading character in the listing should be the letter “c”, indicating that the port is a character device. If there is any other letter in this position, then the port name that was specified is an invalid port name.
- b. The permissions for read and write should be open to everyone (rw-rw-rw-). This is the default configuration for ports on a Solaris machine. If the permissions do not allow for public read and write access, then you should change the permissions.
  - Login or switch user to `root`.
  - Then use `chmod 666 <path_name>`, specifying the complete path that was displayed in the `ck_port_node.sh` listing.

If you find that the permissions are not wide open, then the port is likely to be under the control of the Solaris `port monitor`. This is apt to cause problems again. You should completely disable the LMU port from the port monitor using the system `admintool` utility.

As `root` user, run `admintool` from the system console. Choose 'browse' and then 'serial ports'. Select the LMU port.

Disable the LMU port from the port monitor by clicking on the 'service enable' button.

An alternate method to disable the LMU port from the port monitor is to remove the LMU port record from the file:

```
/etc/saf/zsmon/_pmtab
```

On an AIX machine, run the utility `lsdev -C -c tty` to verify that the specified port is available. If it is not available, then you should use the `smitty` utility to create a new instance of a TTY or change one that has already been defined. Choose 'devices', then 'TTY', then either 'Add a TTY' or 'Change / Show characteristics of a TTY'.

3. Verify that the cable connection is good.

The LMU supports hardware handshaking. In order to assure 100% compatibility between the host and LMU, you will want to verify the following pin connections. (The following assumes a standard DTE pin arrangement for the host.)

For DB-25 to DB-25 connectors:	
Host	9330 LMU
2	2
3	3
4	4
5	5
6	6
7	7
8	8
20	20

For DB-9 to DB-9 connectors:	
Host	9740 LMU
1	1
2	2
3	3
4	4
5	5
6	6
7	7
7	8

For DB-25 to DB-9 connectors:	
Host	9740 LMU
8	1
3	2
20	4
2	3
7	5
6	6
4	7
5	8

For DB-9 to DB-25 connectors:	
Host	9300 LMU
1	8
2	3
3	2
5	7
6	6
7	4
8	5
--	--

- Verify that the LMU is powered up and functional.

Since `testports` and `testlmu` rely on a valid response from the LMU, these tests assume that the LMU is functional. In some cases an incomplete host-LMU transaction can cause the LMU to stop communicating. In such an event, you can IPL the LMU in order to restore communication.

## Troubleshooting a SCSI Connection

To verify the SCSI hardware connection, you can use either of two utilities. If the SCSI port has already been defined to ACSLS then you can use the command `testports`. Otherwise, use `probescsi.sh`, passing the desired port name as an argument. If the test response is positive, then you have verified a good hardware and software connection. If the response is negative, then you should consider each of these possibilities:

## 1. Verify a valid port definition.

Make sure that the port you have specified is `/dev/mchangerX`. On Solaris machines, the X is the target ID of the library device on the SCSI bus. On AIX systems, X is an incremental number, starting with 0 and incrementing with each new mchanger device that is defined.

## 2. Verify a good hardware connection.

On Solaris use *probe-scsi-all*. For AIX, use *diag*. A positive response verifies a good cable connection and all of the hardware between the host and the library.

If the response is negative, check each of the following:

- **Check the cable.** Look closely at the connectors for any bent pins. Make sure the connectors are securely attached to the equipment at each end of the cable.
- **Check for bus compatibility and proper termination.** A single-ended SCSI bus should be populated only by single-ended devices and a single-ended terminator. A differential SCSI bus should be populated only by differential devices and a differential terminator. You should be able to determine the type of host-bus adapter card installed in your system. Make sure that the library device is compatible with the adapter and that it is properly terminated.
- **Check the library device.** The library must be powered on and functional in order to respond to a SCSI inquiry from *diag* or *probe-scsi-all*.

## 3. Verify a valid device node.

**On a Solaris Machine:**

Run `ck_port_node.sh`. Make sure that the listing is a character device (the first character in the listing should be the letter "c"). Make sure that the device path reveals a SCSI connection to the Library and that the target ID in the listing matches the target defined in the operator panel of the library. Also, make sure that the permissions are set for read and write access to all users (`rw-rw-rw-`).

If an mchanger device does not exist, then you can create one as `root` user, using:

```
$ACS_HOME/install/install_scsi_sol.sh
```

If an mchanger device exists, but you suspect that it may have been improperly created, you can remove it and rebuild it. To remove the mchanger device, switch user to `root` and use the following command:

```
rem_drv mchanger
```

You will need to reboot the system in order for the mchanger driver to be unlinked from the kernel.

Now rebuild the mchanger device. Make sure that the library is powered on and functional, and then run:

```
$ACS_HOME/install/install_scsi_sol.sh
```

In most cases, `$ACS_HOME` is defined as `/export/home/ACSSS`. You will need to be root to use this command.

#### On an AIX System:

Run `lsdev -C -s scsi`. Make sure that an mchanger device is available and that it matches the target ID of the library device.

If a matching mchanger device does not exist, then you can create one as root user, using:

```
$ACS_HOME/install/install_scsi_aix.sh
```

In most cases, `$ACS_HOME` is defined as `/export/home/ACSSS`. You will need to be root to use this command.

If an mchanger device exists, but you suspect that it may have been improperly created, you can remove it and rebuild it. To remove the mchanger device, switch user to root and use the following command:

```
rmdev -l mchangerX -d
```

where X is the specific instance of the mchanger you want to remove.

Now rebuild the mchanger device node with the command:

```
$ACS_HOME/install/install_scsi_aix.sh
```

In most cases, `$ACS_HOME` is defined as `/export/home/ACSSS`. You will need to be root to use this command.

## ■ Gathering ACSLS Diagnostic Information

Use the data gathering tool to collect complete ACSLS diagnostic information for StorageTek Software Support.

- Login as `root`
- Change to the `diag/bin/` directory
 

```
cd $ACS_HOME/diag/bin
```
- Gather diagnostic information
 

```
./get_data.sh
```
- This creates a file containing a collection of diagnostic information:
 

```
$ACS_HOME/log/output.tar.Z
```

- After you contact StorageTek Software Support about your issue, please send them the `$ACS_HOME/log/output.tar.Z` file.



# Improving Library Performance



## ■ Overview

This chapter discusses how to maximize your library performance during pass-thru activity between Library Storage Modules (LSMs). ACSLS works to minimize pass-thru activity in several ways. You can enhance performance by using these ACSLS facilities to minimize pass-thru activity. They are:

- Minimizing pass-thru activity between LSMs
- Floating cartridges during a dismount
- Entering cartridges into a library
- Ejecting cartridges from a library
- Maintaining empty storage cells in a library
- Minimizing pass-thru activity with ACSAPI requests and ACSLS commands
- Using these ACSAPI requests and ACSLS commands

## ■ Minimizing Pass-thru Activity Between LSMs

To improve tape library performance, it is important to minimize pass-thru activity from one library to another. When cartridges and drives are in connected libraries, it is always desirable if tape mounts occur when both the drive and the cartridge are in the same LSM (library). Mounts that require fewer pass-thrus result in better performance.

## ■ Floating Cartridges During a Dismount

When cartridges are dismounted, ACSLS tries to avoid pass-thru activity among LSMs by assigning a new home cell whenever the cartridge's old home cell is in a different LSM than the transport. This is ACSLS's default behavior. To disable this feature, use the "Extended Store" feature for an LSM. Cartridges being dismounted are "floated" to a new home cell to avoid movement between LSMs.

## ■ Entering Cartridges into a Library

When cartridges are entered into a library, each cartridge is assigned a home cell in the closest LSM to the CAP. If the LSM with the CAP has empty cells, home cells are assigned in this LSM. If the LSM with the CAP is full, home cells are assigned in the closest LSM with empty cells.

To optimize performance, enter cartridges into an LSM with tape drives that are compatible with the cartridge(s). For example, enter 9940 cartridges into an LSM with T9940B tape drives, not into an LSM with only T9840B tape drives. Otherwise, the 9940 cartridges must be passed-thru to an LSM with T9940B tape drives.

## ■ Ejecting Cartridges from a Library

When cartridges are ejected from a library, the cartridges must be moved to the specified CAP(s).

To optimize performance, eject cartridges to a CAP in the LSM where the cartridges are located. If the cartridges are located in several LSMs, choose a CAP that is close to most of the cartridges.

## ■ Maintaining Empty Storage Cells in an LSM

ACSLs cannot “float” cartridges to an LSM on a dismount if all the storage cells are full. Similarly, when cartridges are entered into a full LSM, the cartridges must be passed-thru to an LSM with empty cells.

To optimize performance, you should identify cartridges that do not need to be kept in the library for StreamLine access and eject those cartridges from the library. You may also want to move cartridges from full LSMs to other LSMs with enough empty cells so that all LSMs have empty storage cells.

**Notes:** Managing the cartridges in a library should be done when the library is not busy with mount requests.

## ■ Minimizing Pass-Thru Activity Using ACSAPI Requests and ACSLS Commands

These ACSAPI requests and ACSLS `cmd_proc` commands help you to avoid unnecessary pass-thru activity between LSMs.

The following applies to both ACSAPI requests from a client to ACSLS, and to ACSLS commands entered using `cmd_proc`. The term “request” is used for both ACSAPI requests and `cmd_proc` commands.

## mount scratch and mount \*

For a specified transport, select a scratch cartridge and mount it. Optionally, select the cartridge from a specified scratch pool and/or with a specified media type. ACSLS selects a compatible scratch cartridge in the closest LSM to the transport. In order to rotate use among all cartridges, the compatible cartridge with the least recent access date is selected within the selected LSM.

### Command:

ACSAPI	mount scratch
cmd_proc	mount * <i>drive_id</i> [ <i>pool_id</i> ] [media <i>media_type</i> ]

- For an ACSAPI request, *drive\_id*, *pool\_id* and *media\_type* must be specified. (Either a specific media type, ALL\_MEDIA\_TYPE, or ANY\_MEDIA\_TYPE can be specified for *media\_type*.)
- With a cmd\_proc mount \* command, if *pool\_id* is not specified, it defaults to the common pool (pool 0).

A specific media type can be identified. If media \* is specified, ANY\_MEDIA\_TYPE is chosen. If media is not specified, ALL\_MEDIA\_TYPE is chosen.

The special *media\_type* values of ANY\_MEDIA\_TYPE and ALL\_MEDIA\_TYPE are processed as follows:

- When ALL\_MEDIA\_TYPE is specified, a cartridge with a media type compatible with the transport is selected. (This is based on the media\_compatibility file.)
- When ANY\_MEDIA\_TYPE is specified, the scratch\_preferences file identifies the preferred list of media to mount on a transport.

## query mount

For a specified cartridge (*vol\_id*), return a list of transports that are compatible with the cartridge's media type, ordered by proximity to the cartridge. (The compatible transports in the closest LSM are listed first, etc.)

**Command:** query mount *vol\_id*

## query mount scratch and query mount \*

For a specified scratch pool, a list of all transports compatible with the media type(s) of the cartridges in a specified scratch pool is returned. A specific media type can be specified to restrict the transports to only those compatible with the specified media type.

The list of transports returned is ordered so the transports closest to the densest scratch pools are listed first.

**Command:**

ACSAPI	<code>query mount scratch</code>
cmd_proc	<code>query mount * pool_id [media media_type]</code>

Either a specific media type or ALL\_MEDIA\_TYPE (media \* in a cmd\_proc command) can be specified. ANY\_MEDIA\_TYPE is **not** supported.

**Note:** When ALL\_MEDIA\_TYPE is specified, cartridges with media compatible with the transport are selected to determine the densest scratch pool.

## ■ Using these ACSAPI Requests & ACSLS Commands

This section describes how to use the ACSAPI requests and commands for improving tape library performance.

In the following discussion the term “request” is used for both ACSAPI requests and cmd\_proc commands.

When mounting a specific cartridge (where the vol\_id is known):

- Precede the mount request with a query mount request.
- Pick the first “available” transport, and specify this transport in the mount request.

When mounting a scratch cartridge on a specific transport, there are two options:

- To select the closest scratch cartridge to a specific drive:

Use a mount scratch request specifying a transport and, optionally, a scratch pool.

For ACSAPI requests, specify one of the following:

- ALL\_MEDIA\_TYPE (compatible media is selected)
- ANY\_MEDIA\_TYPE (uses the scratch preference list).

For cmd\_proc commands, select one of the following:

- Do not specify a media type (compatible media is selected)
- Specify media \* (uses the scratch preference list).

- To select a specific drive from a list of drives and then select the closest scratch cartridge:

Pick the transport, then pick the scratch tape: This ensures the transport is close to the scratch cartridge.

- a. Enter a `query mount scratch` request to identify the closest available transport to the most scratch media in the desired scratch pool.
- b. Enter a `mount scratch` request to mount a scratch cartridge on the selected transport. Optionally, specify the scratch pool.

For ACSAPI requests, either specify:

- `ALL_MEDIA_TYPE` (and compatible media will be selected), or
- `ANY_MEDIA_TYPE` (to use the scratch preference list).

For `cmd_proc` commands, either:

- Do not specify a media type (and compatible media will be selected), or
- Specify `media *` (to use the scratch preference list).



# Firewall Security Option

---



## ■ Overview

The firewall-secure option allows you to run ACSLS behind a firewall while the client software makes requests across that firewall.

Firewall security is also offered to ACSLS clients, which also allows them to operate behind their own respective firewalls. This is made available by StorageTek to its ISVs (Independent Software Vendor partners). Please contact the ISV for your client software component to find out the latest status for each specific client.

## ■ Running ACSLS behind a Firewall

This Firewall-secure solution provides the following capabilities:

- Allow ACSLS to run behind a firewall (i.e. ACSLS on the secure side of firewall, client on opposite side)
- Allow ACSLS client(s) to run behind their own firewall(s) (i.e. client(s) on secure side, ACSLS on the opposite side of firewall).

**Important:** To do this, the ISV must have implemented the supplied changes within their client side software component.

- Preserve compatibility with current ACSLS client implementations, allowing those clients to continue to run with ACSLS in the firewall solution.
- Preserve current ACSAPI/Client functionality and performance. This includes all functionality that is available in a non-firewall environment.

A complete solution would include combining the first two capabilities above. This would allow ACSLS and the ACSLS client(s) to each run behind their own respective firewalls (i.e. two firewalls between ACSLS and the client(s)), and still have the same communications performance as within a non-firewall environment.

## ■ Addressing security areas

ACSLs has addressed the following security concerns as follows:

### RPC

The use of RPC within ACSLS is a concern for some sites in trying to run within a firewall environment. Preserving compatibility with the current installed client base precludes the ability to remove RPC completely from the ACSLS.

However, the ACSLS firewall-secure feature has addressed the concerns inherent in RPC, which are:

- The need to allow outside (untrusted) parties to initiate connections to the trusted host across an unrestricted range of ports (1024-65535).
- The exposure of the mapping of available services on a platform via the portmap (or rpcbind) daemon running on well-known port 111.

### Security

In a firewall solution, the fundamental security comes from restricting access from the untrusted (non-secure) side into the trusted (secure) side. In all cases, some limited and controlled access must be allowed in order to perform communications and allow data exchange. The goal is to allow that data exchange within a well-defined and restricted set of entry points, allowing you to control those access points and their corresponding communications. This goal is met by this solution.

### Communications Components

ACSLs/Client communications rely on two network interface components to handle network communications between client platforms and the ACSLS platform. Software which acts as a client or proxy-server for ACSLS implements one of these two components, in order to be compatible with ACSLS platforms and existing clients. The component residing on the client platform is known as the SSI; the component residing on the ACSLS platform is known as the CSI. While it would be desirable to implement all changes within one side (e.g. the ACSLS platform), in order to maintain client compatibility and to provide all the firewall-secure features it is necessary that corresponding changes be made to each side where the benefits are to be realized. The positive side of this is that each side can independently implement the features and realize the firewall-secure benefits on its side (e.g. changes to the ACSLS allow the ACSLS platform to run behind a secured firewall).

## ■ Benefits of the Firewall-Secure Option

### Server Side

With changes to just the server side component, as provided within this Firewall-secure solution, the benefits are:

- Restricts incoming connections for ACSLS communications to a single TCP port for all registered program numbers (there are two registered program numbers for the ACSLS CSI, both of which will be serviced by one single port).
- Allows users to specify the identity of that port, and configure their firewall in a corresponding fashion.
- Allows users to turn off ACSLS communications to UDP ports.
- Allows users to disable any communication by the ACSLS server to the **client-side** portmapper(s)\* (UDP/TCP port 111). The portmapper must still remain running on client platforms to preserve compatibility with client side code. However, it will not be used for network communications initiated by the server, and therefore the clients' firewall(s) can be configured to disallow access to it.
- Outgoing connections from the ACSLS server side to the client(s) are unrestricted with respect to the server-side ports used to preserve current performance. This follows the widely accepted practice by the security community.

### Server Port Restriction

This solution potentially restricts the number of incoming ports to which any outside party can communicate with the server platform (i.e. initiate network connection) to either one or three ports - the single customer-specified port for ACSLS incoming requests, plus possibly the two portmapper ports (TCP & UDP port 111).

**Note:** To disallow client access to the **ACSLs server** portmapper, and thus disallow access to UDP and TCP ports 111, the changes must be made to the **client** software component. See the client side discussion below.

The server side of the solution, above, is implemented completely within this firewall-secure ACSLS solution.

## Client Side

An identical set of restrictions can be realized on the client side. This would give the client an identical capability of residing behind its own secure firewall, as described in the server side discussion above. These restrictions allows the user to:

- Restrict incoming connections for communications (response) to the client to a single TCP port for each registered program number. There is one registered program number for the ACSLS SSI.
- Specify the identity of this port, and configure the firewall in a corresponding fashion.
- Turn off client-side communications to UDP ports.
- Disable any communication by the client to the **ACSLs server** portmapper (UDP/TCP port 111). The portmapper must still remain running on the ACSLS platform to preserve compatibility with ACSLS code. However, it will not be used for network communications initiated by the clients, and the ACSLS server firewall can be configured to disallow access to it.
- Outgoing connections from the client side to the server are unrestricted with respect to the client-side ports used to preserve current performance. This follows widely accepted practices by the security community.

## Client Port Restriction

This solution restricts the number of incoming ports to which any outside party can communicate with the client platform (i.e. initiate network connection) to between one and three ports - one specified port for client incoming responses, and possibly the two portmapper ports (TCP & UDP port 111).

**Note:** To disallow ACSLS server access to the **client's** portmapper (and thus disallow access to UDP and TCP ports 111), the changes must be made to the **ACSLs server** software component (see server side discussion above).

This solution has a two-step implementation:

- StorageTek has made the needed code changes to the ACSLS CDK source code (CSC Developer's Toolkit)
- Clients of ACSLS who wish to provide this security for their client platform must integrate these changes into their client-side SSI code, rebuild that product, and again certify their CSC (Client System Component) with StorageTek's ACSLS.

## Advantages

Each of the client-side and server-side parts of the solution is independent. This means that if only one of the two sides is behind a firewall with respect to the other side, the solution (software changes) need only be implemented on that side. In addition, that partial solution maintains compatibility with all current client and server implementations which already exist, as well as compatibility with other software components which use the CSI / SSI interface. This includes compatibility with current StorageTek products. This solution does not impact current performance, with respect to client / server communications.

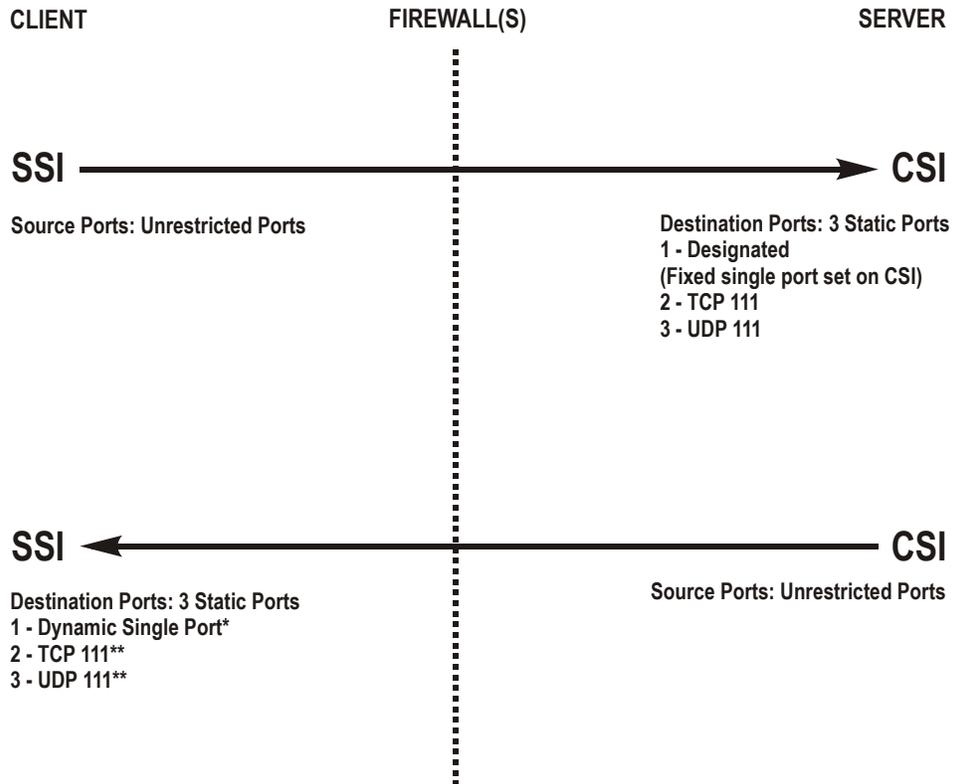
## ■ Firewall-Secure Solution Scenarios

The following diagrams show possible scenarios of the operation, port usage, and relationship of the ACSLS components when used across a firewall. They are intended to be understood in conjunction with the text just presented (above). The “SSI” in the following diagrams is the network interface component of ACSLS that runs on the client side of the communications. The CSI is the network interface component of ACSLS that runs on the ACSLS platform.

## Example 1

In [Figure 63](#), firewall security is implemented on the ACSLS server side (CSI) only.

**Figure 63. Firewall security on ACSLS server side (CSI) only**



S403\_006

### Notes:

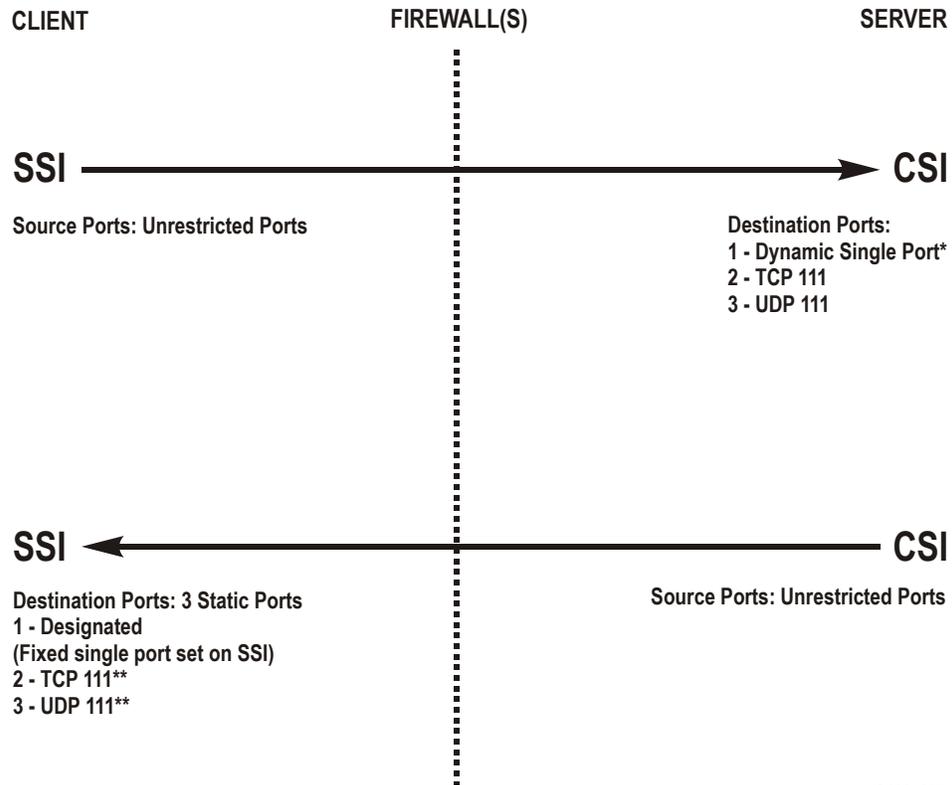
\* Dynamic in this example means that the port is selected by the SSI at startup from the range 1024-65535. The port is not designated by the user, nor is it typically the same port across new executions of the SSI (i.e. from one instance of an SSI running process to the next).

\*\* The portmapper 111 port(s) on the SSI side is **only rarely** queried by the CSI. It is only accessed by the CSI in the case where the return port number provided by the SSI in its request packet does not function (i.e. results in a network interface failure) for sending the response packets back to the SSI. In this case, as a retry mechanism, the CSI queries the SSI-side portmapper for the port to use, which is registered with the portmapper under the SIE's program number.

## Example 2

In [Figure 64](#), firewall security is implemented on the client side (SSI) only.

**Figure 64. Firewall Security on Client Server Side (SSI) Only**



S403\_007

### Notes:

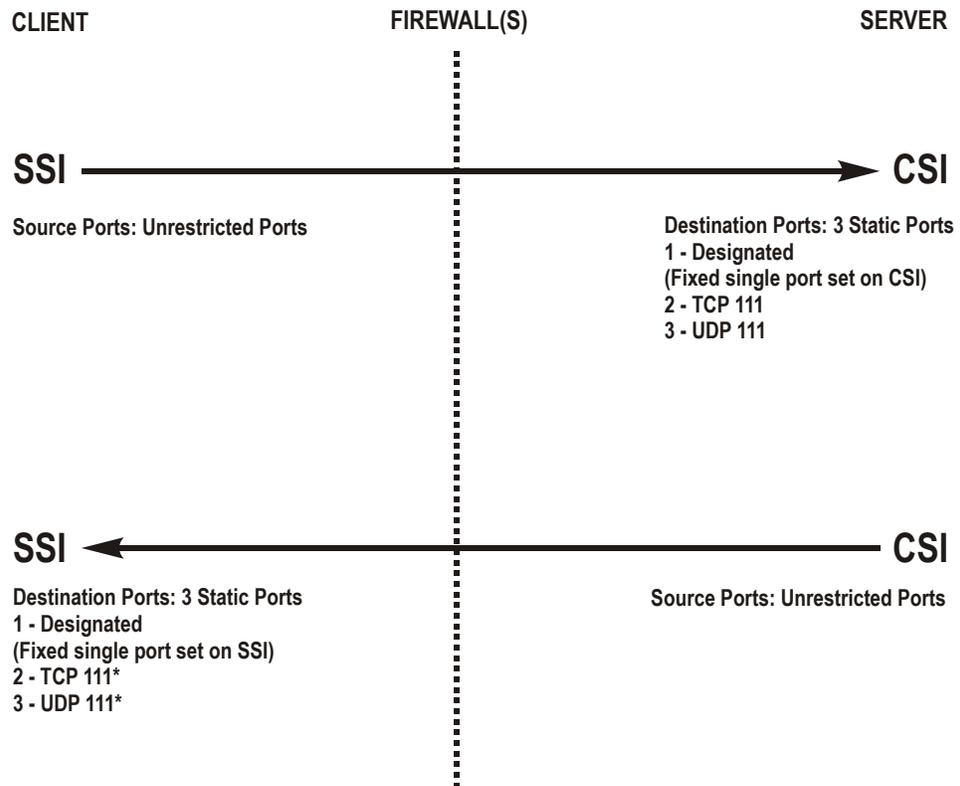
\* Dynamic in this example means that the port is selected by the CSI at startup from the range 1024-65535 and the port is not designated by the user, nor is it typically the same port across new executions of the CSI (i.e. from one instance of an CSI running process to the next).

\*\* The portmapper 111 port(s) on the SSI side is only rarely queried by the CSI. It is only accessed by the CSI in the case where the return port number provided by the SSI in its request packet does not function (i.e. results in a network interface failure) for sending the response packets back to the SSI. In this case, as a retry mechanism, the CSI queries the SSI-side portmapper for the port to use, which is registered with the portmapper under the SSI's program number.

### Example 3

In [Figure 65](#), both client (SSI with CDK 2.2 and Firewall-secure Supplement) and the ACSLS server (CSI) are implementing Firewall-Secure API's. The client and server is still relying on the portmapper for port identification).

**Figure 65. Firewall Security on Client (SSI) and ACSLS Server (CSI) using Portmapper**



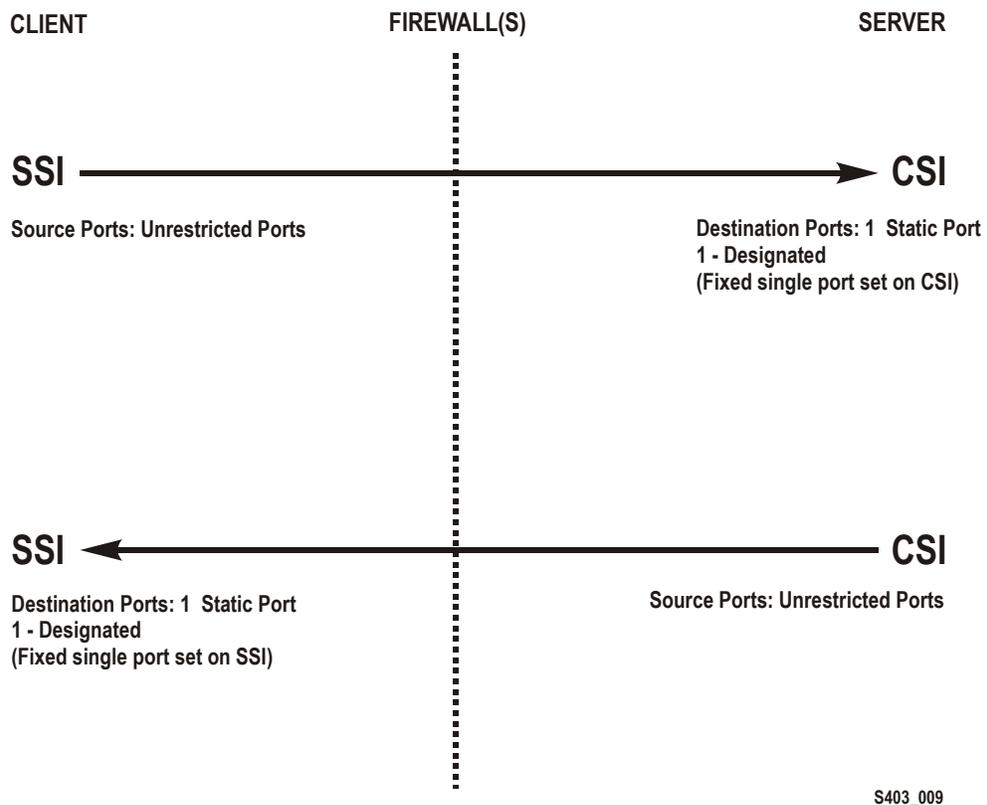
**Notes:**

\* The portmapper 111 port(s) on the SSI side is only rarely queried by the CSI. It is only accessed by the CSI in the case where the return port number provided by the SSI in its request packet does not function (i.e. results in a network interface failure) for sending the response packets back to the SSI. In this case, as a retry mechanism, the CSI queries the SSI-side portmapper for the port to use, which is registered with the portmapper under the SSI's program number.

## Example 4

In [Figure 66](#), both Client ( SSI with CDK 2.2 and Firewall-secure Supplement) and ACSLS Server (CSI) implementing Firewall-Secure API's (Client and Server have implemented the "No Portmapper" feature).

**Figure 66. Firewall Security on Client (SSI) and ACSLS Server (CSI) Without the Portmapper Feature**



## ■ Turning On the Firewall-Secure Option

To turn on the firewall-secure option, you need to set several variables using the `access_config` utility.

1. Log in as `acsss`.
2. At the password prompt, enter the password for user `acsss`.
3. Stop the ACSLS server

**Note:** You must shut down the ACSLS server for the new firewall-secure variables to take effect.

```
kill.acsss
```

4. To run the configuration script, enter

```
acsss_config
```

The ACSLS feature configuration screen appears.

5. Select option 1 - Set CSI tuning variables

Accept the default for all variables except for the following.

- a. Set the value to TRUE at the following prompt:

```
Changes to alter use of the TCP protocol will not
take effect until the product is restarted. CSI
support for RPC using the TCP protocol is enabled
[TRUE].
```

**Variable:** CSI\_TCP\_RPCSERVICE

Turning on TCP insures that the TCP protocol is available for use by clients of ACSLS for network communications. The firewall-secure feature of ACSLS supports TCP only, so clients should perform network communications using this protocol.

- b. Set the value to FALSE at the following prompt:

```
Changes to alter the use of the UDP protocol will
not take effect until the product is restarted. CSI
support for RPC using the UDP protocol is enabled
[TRUE].
```

**Variable:** CSI\_UDP\_RPCSERVICE



**CAUTION:** You must ensure that no ACSLS clients are depending on this UDP protocol. The firewall-secure ACSLS runs on TCP only.

Turning off UDP insures that no clients will access the server using this protocol. This allows you to disallow all general UDP access to the ACSLS platform at the firewall, allowing only those accesses which are specifically required in your environment.

You **must** allow clients access to the UDP and TCP port 111 for portmapper access, unless those clients implement the firewall-secure feature, and specifically turn off their queries to the ACSLS portmapper.

- c. Set the value to NEVER at the following prompt:

```
Changes to alter use of the port mapper will not
take effect until the product is restarted. Enable
port mapper: (ALWAYS / NEVER /
IF_DUAL_LAN_NOT_ENABLED) [IF_DUAL_LAN_NOT_ENABLED].
```

**Variable:** CSI\_USE\_PORTMAPPER

**NEVER** allows clients of ACSLS to disallow external access to the portmapper on those client platforms.

**IMPORTANT:** This does **not** allow you to turn off external access to the portmapper on the ACSLS platform; to do that, the client(s) of ACSLS must have adopted the firewall-secure changes in the client software component(s), and this feature must be turned on in the client software component.

This feature ensures that the ACSLS server will not make any queries of the portmapper on the client platform. This allows any firewall **which is protecting the client** to disallow access to the portmapper.

- d. Set the value to TRUE at the following prompt:

```
Enable CSI to be used behind a firewall (user-
defined inbound port) (TRUE/FALSE) [FALSE]:
```

**Variable:** CSI\_FIREWALL\_SECURE

**TRUE** allows you to specify the single port that ACSLS will use for accepting inbound client communications (i.e. TCP connections). This variable simply enables this feature. The specific port will be specified in the next variable.

- e. Set the value to an available fixed port on the ACSLS server at the following prompt:

```
Port number used by the CSI to receive incoming
ACSLs requests.
```

**Variable:** CSI\_INET\_PORT

This is the port which will be used by the ACSLS CSI component for accepting incoming network connections.

**IMPORTANT:** You must configure your firewall to allow incoming connections on this port. This ensures that only that port is exposed for use by those outside clients wanting to initiate communications with ACSLS. You may disallow connections on all other incoming ports except this one, and UDP/TCP port 111 (unless clients have implemented the feature to eliminate their queries to the ACSLS portmapper; in that case, port 111 may also be disallowed at the firewall). The recommended default value for this port is 30031. It is unlikely (but not impossible) that this port will be used by other processes on most systems. See the [Appendix F, "Troubleshooting"](#) for steps to take if there is a port conflict.

6. Select E to exit `acsss_config`.
7. Restart ACSLS by entering:

```
rc.acsss
```

Your changes are saved.

## ■ Turning Off the Firewall-Secure Option

Some of the variables used above for turning on the firewall-secure feature are also related to turning off that feature. To turn off the firewall-secure behavior, it is only necessary to perform the steps below, but a specific site may wish to make modifications to other variables as well.

1. Log in as `acsss`.
2. At the password prompt, press enter the password for user `acsss`.
3. Stop the ACSLS server

**Note:** You must shut down the ACSLS server for the new firewall-secure variables to take effect.

```
kill.acsss
```

4. To run the configuration script, enter  
`acsss_config`
5. Select option 1 - Set CSI tuning variables

Change the following values that were set when you configured the firewall-secure feature. You must change the following variables:

- a. Set the value to ALWAYS at the following prompt:

```
Changes to alter use of the port mapper will not  
take effect until the product is restarted. Enable  
port mapper: (ALWAYS / NEVER /  
IF_DUAL_LAN_NOT_ENABLED) [IF_DUAL_LAN_NOT_ENABLED].
```

**Variable:** `CSI_USE_PORTMAPPER`

- b. Set the value to FALSE at the following prompt:

```
Enable CSI to be used behind a firewall (user-  
defined inbound port) (TRUE/FALSE) [FALSE]:
```

**Variable:** `CSI_FIREWALL_SECURE`

6. Select E to exit `acsss_config`.
7. Restart ACSLS by entering:

```
rc.acsss
```

Your changes are saved.

## ■ Firewall-Secure Configuration

The following requires that you are knowledgeable with regard to configuring the network firewall behind which ACSLS resides. ALL firewalls are “third-party” software, and will have varying details with regard to setting them up correctly for protecting your network environment. The following is not meant to be a recommendation of firewall security policy, but rather a set of helpful instructions for what the firewall must / can do with regard to the ACSLS product only. See your System Administrator for other security details.

Here is a list of details for setting up your firewall with respect to the ACSLS platform:

- Put in place an overall rule to disallow UDP incoming and outgoing connections.
- Put in place an overall rule to disallow TCP incoming connections (TCP outgoing connections must remain open).
- Put in place a specific rule to allow incoming TCP connections on the port which you specified for the ACSLS usage. **IMPORTANT:** This port **must** match the one you configured under `acsst_config`, or you will receive no client communications at the ACSLS server.

If all of your clients have implemented the firewall-secure feature and make no queries to the ACSLS platform's portmapper, you are done. If the clients still make use of that portmapper on the ACSLS platform, you must add the following:

- Put in place a specific rule to allow incoming and outgoing connections on the well-known portmapper TCP and UDP port 111.

### Example

The following is an example of the rules which were put in place for an iptables-based firewall in order to put all of the above rules in place.

**Note:** These are in addition to other rules configured for the specific firewall.

```
echo " - FWD: Allow all connections OUT and only existing/
related IN"
$IPTABLES -A FORWARD -i $EXTIF -o $INTIF -m state --state \
    ESTABLISHED,RELATED -j ACCEPT

# These rules allow client access to the portmapper
$IPTABLES -A FORWARD -p tcp -i $EXTIF --dport 111 -j ACCEPT
$IPTABLES -A FORWARD -p udp -i $EXTIF --dport 111 -j ACCEPT

# These rules allow client access to the ACSLS CSI for network
communication

# Note: This assumes that the CSI firewall-secure port was
specified as 30031
```

```
$IPTABLES -A FORWARD -p tcp -i $EXTIF --dport 30031 -j ACCEPT
# Catch all rule, all other forwarding is denied and logged.
$IPTABLES -A FORWARD -j drop-and-log-it
```

## ■ Troubleshooting Firewall-Secure Communications

Troubleshooting a network communications interface which includes the ACSLS platform and clients, and now includes intervening firewall(s), may involve multiple steps. By introducing the firewall(s) into the path between ACSLS and its clients, there are more potential causes for network communications failures. Additionally, there are more components that must be configured in a way that corresponds with the settings in other components, and if these settings do not match, the network communications will be impacted. Here is a list of things to check and try if you've done all the configuration work on ACSLS, its client(s), and the firewall(s), and network communications are not working.

1. Checking the ACSLS platform:
  - Is the ACSLS up and running? If not, check the `acsst_event.log` for possible reasons, or for pointers to a possible culprit.
  - Is the CSI being brought up successfully by ACSLS? If not, there should be informative messages in the `acsst_event.log` which point towards the cause. Bad values for some of the configuration parameters or a port conflict are likely possible causes.
  - Is there a port conflict being reported in the `acsst_event.log` which causes the CSI to fail? If so, you should use the “netstat” or similar system utility to tell you which ports are in use on the system, and configure the ACSLS to use an available port. Remember to reconfigure the firewall to match.
  - Is the CSI registering for the port you expect? Use the command `'rpcinfo -p'` to look at the portmap table. The CSI is registered under program number 300031. Check to make sure that the port registered under that program number is the one you expect (the default port is 30031, with one less zero than the program number).
2. If ACSLS and the CSI are up and running and correctly registered, the next step would be to check access to the ACSLS platform across the firewall:
  - Is the ACSLS reachable via basic RPC? Use the “`rpcinfo -t <hostname> <program-number> <version-number>`” command to send a simple RPC request to the CSI. (Use “`man rpcinfo`” on your system to get more information on the `rpcinfo` command and its use.) Do this from a machine on the inside of the firewall with ACSLS (e.g.

from the ACSLS platform itself), and from outside the firewall. If it works from inside but not from outside, then the firewall is intercepting your ACSLS requests. You need to double check the configuration of the firewall and the ACSLS port. Also check to be sure that the portmapper is accessible through the firewall (this test cannot be used from outside the firewall if access to the portmapper is disallowed).

- Do the ports configured for ACSLS and for the firewall match? Double-check these parameters. This is a likely cause of failure in network communications. Aside from the configured values, perform the 'rpcinfo -p' command mentioned above to insure that the CSI is indeed registering with the expected portnumber. If it is not, look in the acsss\_event.log for information about the cause.
- Is the ACSLS receiving requests, but unable to send back responses? If you check the acsss\_event.log and find that the CSI reports many dropped network packets or failures to communicate with network clients, then the client requests are getting in, but the responses are not getting out. Again, this is an indication that they are being blocked by a firewall.

3. If your problems are still not resolved.

The above addresses several levels of things to look for. If these yield no specific answer, it's time to do some lower-level checking to find out where communications are being broken down. The best way to do this is through the use of a network packet sniffer facility, such as 'snoop' under Solaris. Use "man snoop" on your Solaris-based system to get more information on the snoop command and its use.

Similar packet tracing facilities are available on other network-connected systems.

- To utilize this, you will have to do your packet sniffing from locations that show you where the packets are getting to and where they are being lost. This may be from both inside and outside the firewall.
- Additionally, looking at the packet data will be informative. If either side is allowing use of the portmapper, it is likely you will see some PORTMAP packets.
- Also, you should see RPC packets passing between the ACSLS and its clients.
- Finally, looking at the transport-level TCP connection will inform you of the specific ports being used on each side for the connection. This is often critical information to find out where the communications are being stopped.
- More detail on performing these operations is beyond the scope of this manual, but your System Administrator should be able to provide some help in this area.

## ■ Frequently Asked Questions

### **Why do I need the firewall-secure solution for ACSLS?**

The firewall-secure solution allows you to effectively run the ACSLS behind a firewall, and allows you to restrict ports on that firewall so that security is significantly enhanced.

### **What releases of ACSLS will support the firewall-secure feature?**

Only ACSLS 7.0.0 and above support this feature.

### **What is the maximum number of ports I will have to leave open if I use this firewall-secure feature?**

The maximum number of ports on which you might have to allow incoming network connections is three: one for the ACSLS network interface, and two for the portmapper (UDP and TCP 111). Outgoing ports are unrestricted, in accord with accepted industry security practices.

### **What is the minimum number of ports I can leave open?**

The minimum number is one. This is possible if your clients (ISV software) have also implemented the firewall-secure features in their client, and make no queries to the portmapper which resides on the ACSLS platform. When that is the case, the only port that need be open for incoming connections is the one user-specified TCP port used by the ACSLS network interface.

### **Why doesn't the feature use a range of ports?**

There is no architectural advantage to using a range of ports, and there are some security disadvantages. The non-firewall-secure ACSLS uses a range of ports which consists of the full range of dynamic ports available on any given platform. This is correctly perceived as a potential compromise to the security of a site. Restricting this as much as possible, without adversely affecting ACSLS performance, is the goal in order to eliminate that compromise. Since the ACSLS network interface uses only one incoming port at any given time, there is no reason to extend the range beyond one port, provided that port is dedicated to ACSLS use for the ACSLS platform.

### **What if the port I choose conflicts with another usage of that port on my system?**

This is one of the reasons that the port is made user-specifiable. The specific ports available will vary from one customer site to another. The user is not allowed to use one of the well-known reserved ports from 0-1023. The default port of 30031 falls within the range of registered ports, which makes it less likely (though not impossible) that another application which uses dynamic ports will utilize it. Although it is in the range of

registered ports, there is no application registered to use it, which makes it a reasonable default selection.

**Does this feature allow me to protect my ACSLS server with a firewall?**

Yes, with this feature in place, your ACSLS server can be put on the trusted side of a firewall, with clients accessing it from the opposite (untrusted) side or from the same side.

**Does this feature allow me to protect my ACSLS clients (ISV components) with a firewall?**

Potentially, yes, but not by itself. In order to realize this scenario, your client software components (clients of the ACSLS) must have adopted the firewall-secure feature, which has been made available via the StorageTek CDK (CSC Developer's Toolkit). Contact your client software provider for a current update on their status.

**If I want to be able to protect my clients with a firewall, what should I do?**

You should contact your client software provider. They can tell you whether they have adopted any firewall-secure changes in their CSC (client software component).

**What about the portmapper? Can I completely disallow access to the portmapper?**

If your clients have adopted the firewall-secure changes, they may allow you to shut off the client's queries to the ACSLS platform's portmapper. In that case, you may disallow access to the portmapper on the firewall which protects the ACSLS platform. In any other case, the clients will depend on the ACSLS server side portmapper to help them make a connection with the ACSLS network interface, and it must be available for their use.

**Why must the client implement some changes in order for my ACSLS server firewall to shut down access to the ACSLS platform portmapper?**

Because it is the client that is making these queries of the ACSLS platform. If the client continues to make these queries, the ACSLS platform must continue to provide the portmapper's services in order for those queries to succeed.

**I think the portmapper is bad. Why didn't you remove it completely?**

The portmapper provides an important service to legacy clients. Removing it completely would invalidate the interface on which those clients depend. In short, no legacy clients would work without recoding, retesting, and again certifying with the new non-portmapper interface. In this firewall-secure solution, we have provided the capability to remove the queries to the portmapper from both the ACSLS to the client, and from the client to the ACSLS, but we cannot force client software to conform to this. Thus, the portmapper must remain available at least as an optional service until

a site's clients have adopted the firewall-secure features and no longer make use of the portmapper service.

**Some of my clients have adopted the firewall-secure features and some have not. How can I take advantage of this?**

Those clients which have adopted these features may be protected behind their own respective firewalls. In addition, access to the portmapper's well-known ports may be restricted at the firewall, and then configured to allow access to the portmapper only by those clients who require it. The details and ability to do this will vary based on the specific firewall in use at the site.

**I think RPC is bad. Why didn't you remove it completely?**

The ACSLS network interface has been RPC-based since the first release of ACSLS. It has proven to be an effective, stable, and reliable mechanism, offering various advantages at the network communications layer. However, it can also be more difficult to secure a platform which uses RPC, due to its common dynamic allocation of ports and use of the portmapper. In this firewall-secure solution, both of these areas are addressed, which allows the customer to effectively configure a firewall in a restricted fashion, yielding the security benefits for which they have the firewall in place.

Additionally, complete removal of RPC from the ACSLS network interface would invalidate all current (legacy) ACSLS clients, making it impossible for any of them to communicate with ACSLS without recoding, retesting, and again certifying their CSCs (client software components).

**How will the firewall-secure feature affect network communications performance and timing between my ACSLS clients and the ACSLS server?**

There is no effect on performance due to the new firewall-secure features. The usage of a firewall may have performance implications, but this will be based on the operational characteristics of each specific customer's firewall implementation. With a firewall which has negligible impact on performance, the ACSLS and its clients will continue to perform as they did prior to installing the firewall-secure feature. Also, the ACSLS network interface tolerances can be configured, so that delays imposed by the firewall could be handled gracefully.

**How does the firewall-secure feature affect the rest of my ACSLS operations?**

There is no effect or impact on other parts of the ACSLS operations due to the installation of the firewall-secure solution.

**How does the firewall-secure feature affect the ACSLS functionality that my clients use (via the ACSAPI)?**

The full set of functionality that is provided through the ACSAPI (and which our ACSLS clients use today to interface with ACSLS) will operate the same under the firewall-secure feature as it does without the feature installed. In particular, this firewall-secure feature supports access control, and also all of the newer features that have been added to the ACSLS product. The full functionality of the ACSAPI will continue to be supported by this feature.

**Does the firewall-secure feature work with the ACSLS HA (High Availability) solution?**

The firewall-secure feature does not adversely affect HA operation. However, the HA solution is not designed to be run across a firewall (i.e. with each HA server on opposite sides of a firewall). The HA solution requires remote access to the portmapper, so the firewall could not disallow that access if an attempt were made to run each server on opposing sides of a firewall. There are other details of running across a firewall that could adversely affect an HA setup; it is highly recommended that this not be done.

If the HA servers are set up on the same secured side of the firewall, that set of HA servers could be set up with the firewall-secure feature, and clients on the opposite side of the firewall would be able to interact across the firewall with those servers with the same performance and behavior as they would against a non-firewall-secure HA solution.

**Does this firewall-secure feature work with other StorageTek software products?**

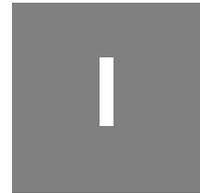
Interoperability with other StorageTek products, as well as partner products (such as client software components which communicate with ACSLS) has been completely preserved. Those products can continue to operate without modification, communicating with the ACSLS server, with the ACSLS server running behind a secured firewall, or in the same environment with those products (as it does today).

**Do other StorageTek software products have the same firewall-secure features?**

Other StorageTek products do not gain the firewall-secure benefit simply by being used in the same environment with the firewall-secure ACSLS. Each product can work with a firewall secured ACSLS (see previous question), but putting each of those products behind its own respective firewall is a question that the specific product itself must address. Some StorageTek products already have built-in policies which allow some restriction at a firewall used to protect the platforms where those products run. Additionally, any product which acts as a client to ACSLS has the option of adopting the firewall-secure changes which were made to ACSLS, and which are provided as part of the StorageTek CDK (CSC Developer's Toolkit).



# Media Management



---

This chapter discusses how to manage the same type of media recorded in different densities.

## ■ Overview

### Problems

Often, new tape drives are introduced that use existing tape media, but record at a higher density. The new drives can often read tapes recorded in the old density, but they cannot write at the old density. The older tape drives usually can neither read nor write in the higher density.

Because of these restrictions, you can encounter these problems:

- When a tape written in the new format is mounted on an older drive, the old drive cannot read the tape.
- Storage management applications often fill up partially used tapes by appending additional files at a later time. This will fail if the tape was written in a different format than the drive uses to record data. When a storage management application tries to append additional files to a tape that was originally initialized in an incompatible format, the mount may fail. In addition, if you load the wrong cartridge in a drive, there is the possibility of overlaying data already on the cartridge.

If you have a mix of both older and newer tape drives in your libraries, you will need to manage your tape cartridges that are the same media type. This document describes how ACSLS pools can be used to separate cartridges written in the old format from cartridges written in the new format.

### Examples

The following examples show tape drives that record on the same media in different formats:

- **9940 Media that is used by both T9940A and T9940B Tape Drives**

The T9940B tape drive uses the same media as the T9940A, but writes data at double the T9940A's density. The T9940B can read T9940A media and can reformat it for writing double density data, but it cannot append data to a previously written T9940A cartridge.

The T9940A drive can reformat a T9940B cartridge for writing single density data, but can neither read from nor append data to a T9940B cartridge.

- **9840 Media that is used by both T9840A & T9840B and T9840C Tape Drives**

The T9840A and T9840B tape drives use the same media and record in the same format. The T9840C also uses the same media, but records at double the density. Customers must manage 9840 media when both T9840A/T9840B and T9840C drives are present.

- **SDLT Media that is used by both SDLT 220 and SDLT 320 Tape Drives**

The SDLT 220 and SDLT 320 tape drives both use the same media, but record in different densities. You must manage the common SDLT media when both types of SDLT drives are present.

## Solution

ACSLS has tools that you can use to manage common media that two or more types of drives write on but are in incompatible recording techniques. The client application must use these facilities to manage the read/append data incompatibilities.

You can manage drives that record on common media in different formats within an ACS by either:

- Replacing all of the older drives in an ACS with the new drives at the same time.

This is the simplest and safest strategy.

Do not append files to any tapes written in the old format after replacing the old drives with the new drives. (With Veritas NetBackup, this is done by suspending the tapes.)

- Gradually replacing the older drives with the new drives.

This requires you to manage the common media recorded in different formats by creating separate pools for each format. These can either be managed with ACSLS pools or by using the facilities of a backup application (for example, Veritas NetBackup, Legato NetWorker, IBM Tivoli, or CA BrightStor).

## ■ Managing Common Media Recorded in Different Densities with ACSLS Pools

The example below describes how to manage 9940 media recorded in both T9940A and T9940B formats. This procedure applies to all cases where different tape drives record on a common media in different formats. Use the following table to apply the procedure to your specific case:

Old Format Drive(s)	New Format Drive(s)
T9940A	T9940B
T9840A or T9840B	T9840C
SDLT 220	SDLT 320

**Note:** The discussion below uses the following syntax conventions for commands and utilities:

- Commands and Utilities entered as is are in **bold**.
- Variables (you must enter the exact value) are in **bold italics**.

Perform the following:

1. Before installing T9940B drives in an ACS that has T9940A drives, identify all 9940 data (non-scratch) cartridges in the ACS and assign them to a T9940A pool. Assign 9940 scratch cartridges to either the T9940A or T9940B pool.

- a. Define T9940A and T9940B media pools:

```
define pool pool_id
```

- b. Report all 9940 media.

You can do this by using either the `display volume` command (option 1) or a custom `volrpt` (option 2) as shown below:

Results are written to a file.

- Option 1

Use the `display volume` command to write the results to a file.

```
display volume * -media STK2P > filename
```

where:

All 9940 tape cartridges (media STK2P) are reported.

*filename* is the filename to which the output is written. The **ACS ID** in which the cartridge is located is also listed.

Read the output, selecting cartridges from specific ACS(s), if desired.

If you want cartridges from one ACS, use the `-home` operand to only select cartridges from this ACS.

```
display volume * -home acs_id.*.*.*.* -media
STK2P> filename
```

- Option 2

Use the custom `volrpt` for all volumes in the ACS. The results are written to a file:

```
volrpt -d -f custom_volrpt_file -a acs_id > filename
```

where:

*custom\_volrpt\_file* is the filename specifying the fields reported in a custom `volrpt`. The following fields are reported:

VOLUME_ID	6	2
MEDIA_TYPE	7	2
VOLUME_TYPE	4	2

*acs\_id* is the ID of the ACS that you are managing.

*filename* is the filename to which the output is written.

Read the output, selecting only the volumes with a media type of STK2P.

- c. Assign the selected volumes to the appropriate pools.

Assign all non-scratch (**VOLUME\_TYPE = "D"**) cartridges to the T9940A pool.

```
set scratch off pool_id vol_id
```

Assign the scratch cartridges (**VOLUME\_TYPE = "S"**) to either the T9940A or T9940B pools.

```
set scratch pool_id vol_id
```

2. After the T9940B drives are installed, use the pools to mount cartridges on compatible tape drives. Again, cartridges written at T9940A density can be read by either T9940A drives or T9940B drives, but only T9940A drives can append data on 9940A media. Only T9940B drives can read or append to cartridges written at 9940B density.

The following commands can be used to identify, select and mount cartridges from the pool appropriate for a T9940A or T9940B drive. Use the pools to ensure you are mounting cartridges on the correct drive.

If you want to mount a scratch cartridge on the correct drive type, follow procedure A. If you want to mount a non-scratch cartridge with data already recorded on it, follow procedure B.

### Procedure A - Mounting Scratch Cartridges

- a. Query drive(s) to identify drive types.

```
query drive drive_id | all
```

- b. Identify the correct pool (media type) for the cartridges to be mounted on the desired drive
- c. Mount scratch cartridges to a drive from the specified pool.

```
mount * drive_id pool_id
```

### Procedure B - Mounting Non-scratch Cartridges

- a. Display the status of drives compatible with a cartridge.

```
display volume vol_id -f pool
```

The pool for this volume is displayed.

- b. Use the following query command to identify a compatible drive.

```
query mount vol_id
```

Drives compatible with a cartridge are displayed. You must select the correct drive type before mounting the cartridge. This query command returns both T9940A and T9940B drives (because both drive types are compatible with 9940 media). Use drive type to separate T9940As from T9940Bs.

- c. Mount the cartridge on the selected drive.

```
mount vol_id drive_id
```

3. When all the data on the cartridge has expired, you can migrate the scratch cartridges to the pool for the new recording format. Because T9940A and T9940B drives can re-initialize cartridges in their density, scratch cartridges can be reassigned to a different pool.

```
set scratch pool_id vol_id
```

4. After all T9940A drives are converted to or replaced by T9940B drives, all 9940 media can be used by T9940B drives as long as data is not appended to cartridges written in T9940A format.



# Installing AIX from a Backup

J

---

The AIX operating system can be installed from a system backup tape created using `smitty mksysb`.

## ■ Installing AIX from a System Backup

**To install AIX from a system backup:**

1. Make sure that the tape drive is turned ON.
2. Make sure that the server is turned ON.
3. Open the tape drive door.
4. Turn the key to "Service".
5. Insert the AIX Operating System backup tape into the tape drive. Close the tape drive door.
6. On the server, press the "Reset" button twice.
7. If the `TESTING COMPLETED` screen displays, press `<Enter>` to continue.

**Note:** For a few minutes the system might appear idle. Do not open the tape drive door. Wait for the next screen to display.

8. The following message displays:

Please define the System Console.

Press `[[F1]]` to define the system console and then press `<Enter>`.

The `INSTALLATION AND MAINTENANCE` screen appears.

9. **Select** Install a system that was created with SMIT "Backup The System" function or the "mksysb" command. Press `<Enter>` to install the operating system from the backup tape.

The `CURRENT SYSTEM SETTINGS` screen displays.

10. Verify that the system settings are correct. If the correct settings are displayed, **select** Install a SMIT "Backup The System" image with the current settings. Press `<Enter>`.

The `FINAL WARNING` screen displays.

11. **Select** Continue with installation. Press `<Enter>`.

12. Press <Enter> to start the tape.

The installation takes 45 minutes to 1.5 hours.

13. Turn the key to “Normal” *before* the installation completes.

When the installation is complete, a screen displays indicating that the AIX Base Operating System installation is complete.

14. Remove the AIX Operating System backup tape from the tape drive.

15. Press <Enter> to reboot the server.

16. During rebooting ignore the following error messages:

```
The System Resource Controller daemon is not active.  
Machine not identical to previous configuration.  
Shutdown, rebooting.
```

**Note:** If the system used to create the backup tape is not the same as the system on which it is now being installed, the server might reboot two or three times.

Each time the server reboots, the system reconfigures. When the server reboots successfully, a login prompt displays.

---

This chapter describes the Client Server Communications Interface (CSCI) component.

This chapter covers the following topics:

- Overview of CSCI
- Error messages
- Environmental variables

Installation of CSCI should follow installation of the operating system and installation and configuration of ACSLS.

**Note:** For ACSLS to function properly, the `START_CSCI_Process` configuration variable must be set to `TRUE`. You do this in Option 1 of `acsss_config`. See [“ACSLs Configuration” on page 115](#).

## ■ Overview of CSCI

The Client Server Communications Interface (CSCI) is a stand-alone component responsible for the passing of request and response packets between a client system and a server system. CSCI is an optional component that lets MVS or RMLS/CSC clients communicate with ACSLS. If you do not use CSCI, you use CSI by default to communicate with ACSLS.

## CSCI Connectivity

CSCI provides IPC and TCP/IP connectivity to link the ACSLS server with MVS clients or RMLS/CSC clients

## Architecture and Subsystems

[Figure 67 on page 556](#) shows an overall CSCI architecture and subsystems.

**Figure 67. CSCI Architecture and Subsystems**

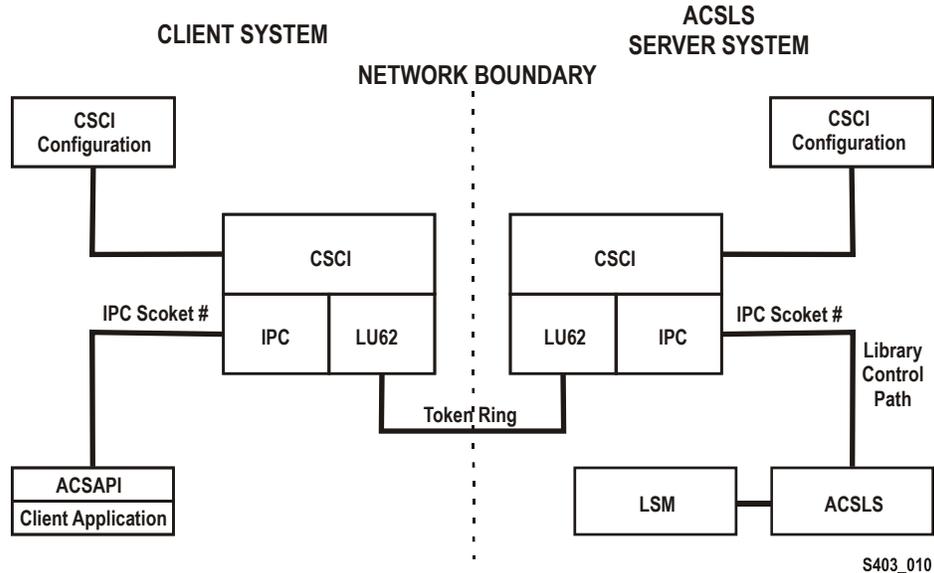
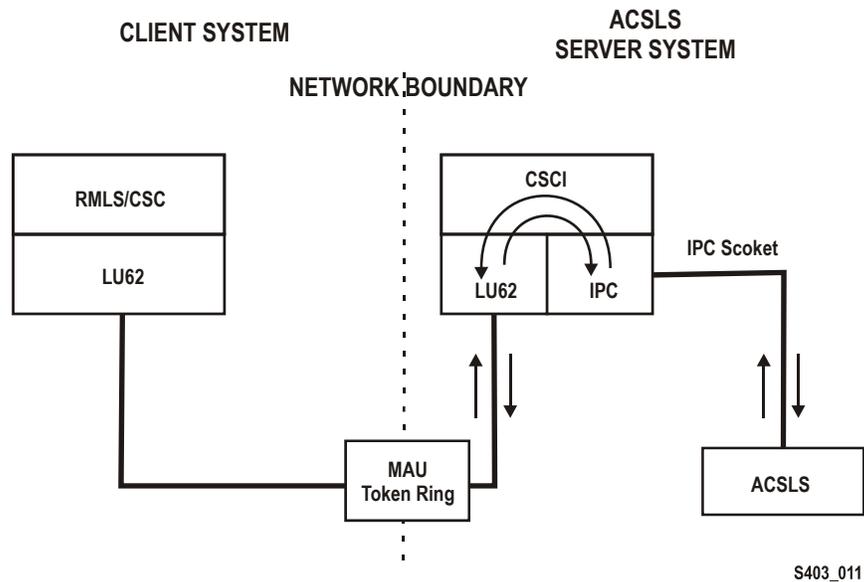


Figure 68 shows how the CSCI system connects with the network interface. On the server side, the data flows from the network hub through the TCP/IP interface to the CSCI interface. From here the data flows through an IPC interface to the ACSLS software in both directions but in a half duplex mode, i.e., one direction at a time.

**Figure 68. CSCI System with Token Ring Interface**



## ■ Error Messages

CSCI installation and configuration error messages relate to the following conditions:

- Improper ownership
- Improper access permission
- Undefined environment variables
- A variety of miscellaneous conditions
- Improper CSCI and TCP/IP configuration

### Common Error Messages

Communication failure, verb = *cmroutine*, rc = *xx*, errno = *yyy* IPC failure on Socket CSCI*zzzzz*

**Explanation:** The read failed on the named connection.

**Variable:**

- *xx* is IBM CPI return code
- *yyy* is the UNIX errno value in */usr/include/sys/errno.h*
- CSCI*zzzzz* is the socket number that was in use
- *cmroutine* is the IBM CPI routine name that failed

Read failed for connection: *connection\_name*

**Explanation:** The read failed on the named connection.

**Variable:** *connection\_name* is the name of the connection on which the read failed.

Write failed for connection: *connection\_name*

**Explanation:** The write failed on the named connection.

**Variable:** *connection\_name* is the name of the connection on which the write failed.

## ■ Environmental Variables

CSCI is invoked by `csciServer.sh`, a Transaction Program Name (TPN) in the directory where ACSLS executables reside. This shell script defines CSCI environment variables.

Table 36 shows descriptions of CSCI environment variables for the server subsystem. Following this table is a shell script like the one provided in the distribution of ACSLS/CSCI software. This shell script is started by TCP/IP on the server.

**Table 36. CSCI Server Subsystem Environment Variables**

Name	Description
CSCI_TRACE	This variable is used to indicate whether CSCI tracing is <b>ON</b> or <b>OFF</b> .
CSCI_SLEEP_TIME	This variable is used to indicate the PER sleep time at the end of each round-robin loop through the connection table (polling timeout). Default is <b>100 msec</b> .
CSCI_INITIAL_CONNECTION	This variable is used to indicate the CSCI's initial connect name to open first. This is the connection service name requiring i/o on initial startup of the CSCI. Clients usually open IPC first; Servers open TCP/IP first.
CSCI_SERVER_NAME	This variable is used to indicate the name of the CSCI Server. It must be set the same for all CSCI's Clients defined to a single CSCI server. <b>The server name must match its CSCI Clients CSCI_SERVER_NAME.</b> The matching of the Server and Client CSCI_SERVER_NAME provides an end-to-end logical link between machines. Finally, this name is used later for the "connectionname" in other environment variables. The value assigned to the CSCI_SERVER_NAME must be replicated in the connectionname part of the subsequent variables.
CSCI_connectionname_NET_TYPE	This variable is used to indicate what Network type this CSCI is configured for from the CIF subsystem. The NETTYPE defines the transport layer used by this CSCI. Valid assignments are <b>LU62</b> or <b>IPC</b> .
CSCI_connectionname_INPUT_SOCKET	This variable is used to indicate the input socket number to use as input to this CSCI. This number would match the Server or Client application output socket number.

**Table 36. CSCI Server Subsystem Environment Variables**

Name	Description
CSCI_connectionname_CON_TYPE	This variable is used to indicate this CSCI type. This CON_TYPE defines the CSCI to be either a <b>SERVER</b> or a <b>CLIENT</b> .
CSCI_connectionname_TRANSLATE_TYPE	This variable is used to indicate this CSCI XDR type to be employed during execution. This can be assigned to either <b>XDR</b> or <b>NONE</b> .
CSCI_connectionname_DESTINATION_SOCKET	This variable is used to indicate destination or output socket number to be used by the CSCI Server. This socket number would be the matching input socket number to the CSCI Server Application.



# Glossary

---

## A

**absent cartridge** A cartridge that is in the database, but that couldn't be found when all recorded locations for the cartridge were catalogued. If a nonzero retention period is set, the volume status is changed to STATUS\_VOLUME\_ABSENT.

**ACS** See Automated Cartridge System.

**ACSEL** See ACS Event Logger.

**ACS Event Logger (ACSEL)** The software component that receives messages from other ACSLS components and writes them to an Event Log.

**ACS ID** A unique identifier for an ACS.

**ACSLH** See ACS Library Handler.

**ACS library** A library is composed of one or more ACSs, attached tape drives, and cartridges residing in the ACSs.

**ACS Library Handler (ACSLH)** The part of the ACSLM that communicates directly with the LMU.

**ACSLM** See ACS Library Manager.

**ACS Library Manager (ACSLM)** The software component that validates and routes library requests and responses.

**ACSLs** See ACS Library Software.

**ACSLs database** ACSLS database containing information about the location and status of the data or cleaning cartridges. The information includes cell location, scratch status, etc.)

**ACSLs platform** The server hardware and software that provide the proper environment for ACSLS.

**ACS Library Software (ACSLs)** Manages ACS library contents and controls ACS library hardware to mount and dismount cartridges on ACS cartridge drives.

**ACSLs database** A database used by ACSLS to track the library configuration and the locations and IDs of all data or cleaning cartridges in the library.

**ACSSA** See ACS System Administrator.

**ACS System Administrator (ACSSA)** The interface between the Command Processor and the rest of the system.

**ADI** Application Data Interchange.

**audit** A physical inventory of the contents of all or part of a library.

**Automated Cartridge System (ACS)** The library subsystem consisting of a single or dual LMU, and 1 to 24 LSMs connected to that LMU.

**automated library** See library.

**beginning of tape (BOT)** The location on a tape where written data begins.

## B

**BOT** See Beginning of Tape.

## C

**CAP** See Cartridge Access Port.

**CAP ID** A unique identifier for the location of a CAP. A CAP ID consists of the ACS ID, the LSM number, and the CAP number.

**cartridge** A plastic housing containing a length of data recording tape. The tape is threaded automatically when loaded in a transport. A plastic leader block is attached to the tape for automatic threading. The spine of the cartridge can contain an OCR/Bar Code label listing the volume ID.

**Cartridge Access Port (CAP)** A bidirectional port built into the door panel of an LSM, which provides for the manual entry or automatic ejection of data or cleaning cartridges.

**cartridge drive (CD)** A device containing two or four cartridge drives and their associated power and pneumatic supplies.

**cartridge tape I/O driver** Operating system software which issues commands (e.g., read, write, and rewind) to cartridge subsystems.

**cartridge transport** An electromechanical device that moves tape from a cartridge over a head that writes and reads data from the tape. A transport is distinct from the power and pneumatic sources that supply the electricity and air it needs to function. See cartridge drive.

**CCI** See client computing system.

**CD** See cartridge drive.

**cell** A receptacle in the LSM in which a cartridge is stored.

**channel** A device that connects the host and main storage with the input and output control units.

**client applications** Software applications that manage tape cartridge contents. They access tape cartridges by interacting with ACSLS. Any number of client applications can be resident on a client system.

**client computing system** A computer and an executable image of the operating system.

**client software** This software manages tape cartridge contents, generates requests for cartridges, and drives data to and from cartridges. The client software is not part of ACSLS.

**Client System Component** Software which provides an interface between the client computing system's operating system and ACSLS.

**Client System Interface (CSI)** The software component that translates and routes messages between the ACS Library Manager and the Client System Component.

**command access control** Limits access to commands.

**command area** The bottom area of the cmd\_proc interface where you enter requests and receive responses.

**command processor (cmd\_proc)** The screen interface of the ACSLS. cmd\_proc lets you enter the commands described in Chapter 7.

**control path adapter** A hardware device which converts a Client Computing System's control protocol to the control protocol of the StorageTek Library Control System.

**control unit (CU)** A microprocessor-based unit logically situated between a channel and up to sixteen cartridge transports. The CU translates channel commands into transport commands and sends transport status to the channel.

**CSE** Customer Services Engineer.

**CSC** Client System Component.

**CSI** See Client System Interface.

**CSI variables** Used to define various options to fine-tune communications

between a CSC and the CSI. You change these variables in the `acsss_config` program.

**CU** See control unit.

**cycle error messages** Messages that indicate a library or ACSLS failure.

## D

**database** A collection of interrelated data records. See also ACSLS Database.

**data path** The network path that allows client applications read/write access to tape cartridges.

**data path adapter** A hardware device which translates a Client Computing System's data protocol to the data protocol of the StorageTek Control Unit.

**display area** The top area of the `cmd_proc` interface that collects messages regarding the status of the library.

**dynamic configuration** allows you to implement configuration changes to ACSLS libraries (and components) while ACSLS remains online and running.

## E

**ejected cartridge** A cartridge that has been ejected from the library. If a nonzero retention period is set, the cartridge status is changed to `STATUS_VOLUME_EJECTED`.

**end of tape (EOT)** The location on a tape where written data ends.

**EOT** See end of tape.

**EPO** Emergency Power Off.

**EPROM** See erasable programmable read only memory.

**erasable programmable read-only**

**memory (EPROM)** A special memory chip that can be erased and reprogrammed.

**Event Log** A file, maintained by the ACSEL, that contains messages describing library and ACSLS events.

**Event Logger** See ACS Event Logger.

**external label identifiers** A six-character alphanumeric label on the outside edge of a cartridge used to identify a physical tape cartridge. It may consist of uppercase letters A through Z, numerals 0 through 9, \$, #, and blanks.

## F

**full installation** A complete software installation required for new customer sites or for existing sites where a new library has been installed.

## H

**HLI** Host/LMU Interface. One way that the StreamLine Library Manager communicates with a library.

**HLI-attached** Libraries that are connected to the StreamLine Library Manager through the HLI. These libraries can be connected through a serial interface (serial-attached) or through a TCP/IP interface (TCP/IP-attached).

**home location** The cell associated with a given cartridge.

## I

**ID** Identifier or identification.

**Informix** The relational database used by ACSLS 6.1.

**Informix Storage Manager (ISM)** The Informix database manager, which manages

database backups and offloaded transaction logs.

**Initial Program Load (IPL)** A process that activates a machine reset, initiates wake up diagnostics (from EPROMs) and loads functional code.

**inline diagnostics** Routines that test components of a subsystem while operating on a time-sharing basis with the functional microcode in the subsystem component.

**in-transit cartridges** Cartridges between their source and destination locations. Cartridges are considered in-transit if they are in pass-thru ports, robot hands, or playground.

**I/O** Input/Output.

**IPC** Interprocess Communication.

**IPL** See Initial Program Load.

**ISM** See Informix Storage Manager.

## J

**journal** A sequential log of changes made to the database since the last checkpoint.

## L

**LAD** Lock Access Door.

**LAN** See local area network.

**large CAP (LCAP)** A 40-cartridge CAP with the storage cells arranged in four removable magazines of ten cells each. The magazines appear as a single column of 40 cells to the host software.

**LCAP** See large CAP.

**LCU** See Library Control Unit.

**LED** See Light Emitting Diode.

**library** A library is composed of one or more ACSs, attached tape drives, volumes in the ACSs, and the ACSLS software that controls and manages the ACSs.

**library configuration options** Allows the customer to specify the number of ACSs in the library and the connections between each ACS and the server system.

**library control component** Software which controls the mounting and dismounting of cartridges in the ACS.

**library control processor** Properly configured computer hardware that, with the addition of appropriate software, supports the operation of the Library Control Software.

**library control system** The library control platform loaded with library control software (ACSLs).

**library control software** The software components of ACSLS including the library control component, the Client System Interface and Library Utilities.

**Library Control Unit** The portion of the LSM that controls the picking, mounting, dismounting, and replacing of data and cleaning cartridges.

**library drive** A cartridge transport attached to an LSM that is connected to, and controlled by, a client system. Library drives interact with the LCU during automated tape cartridge mount and dismount operations. Library drives interact with a client application during tape data transfer operations. Library drives are individually addressable by the ACSLM and are individually accessible by client applications. See Cartridge Transport.

**library errors** Errors that occur because the library is offline, has suffered hardware failure, is unavailable, etc.

**Library Management Unit (LMU)** The portion of an ACS that manages LSM's, allocates their resources, and communicates with ACSLS.

**Library Storage Module (LSM)** An ACS structure that provides the storage area for cartridges, cartridge drives, CAPs, and the robot necessary for moving them.

**light emitting diode (LED)** A light emitting device that uses little energy and is used mainly to indicate on/off conditions.

**LMU** See Library Management Unit.

**local area network (LAN)** A computer network in which any component in the network can access any other component. This is the type of interface between an LMU and attached LSM's.

**LSM** See Library Storage Module.

**LSM ID** A unique identifier for an LSM. The LSM ID consists of the ACS ID and the LSM number.

## M

**missing cartridge** A cartridge that is in the database, but couldn't be found. If a recorded possible location for the cartridge could not be examined due to an offline LSM or a drive not communicating, the cartridge is marked MISSING instead of ABSENT. The cartridge status is changed to STATUS\_VOLUME\_MISSING.

## N

**network adapter** Equipment that provides an electrical and logical interface between a network and specific attached equipment.

**Network Interface (NI)** An interface between the server system and the client systems that maintains network connections and controls the exchange of messages.

The NI is resident on the server system and each client system.

**NI** See Network Interface.

## O

**OCR** Optical character recognition.

**ONC** Open network computing.

**Open Systems Interconnection (OSI)** A software architecture model of the International Organization for Standardization. The OSI model provides standards for the interconnection of data processing systems.

**OSI** See Open Systems Interconnection.

**OSLAN** Open Systems Local Area Network.

## P

**Pass-Thru Port (PTP)** Mechanism that allows a cartridge to be passed from one LSM to another in a multiple LSM ACS.

**PCAP** See priority CAP.

**playground** A reserved area of special cells (within an LSM) used for storing diagnostic cartridges and cartridges found in-transit upon power-on and before initialization of the LSM is completed.

**policy** The policy files are located in \$ACS\_HOME/data/external. These miscellaneous files define your policies for ACSLS. They specify Access Control settings, scratch preferences, Extended Store LSMs, custom volrpt settings, cartridge attributes (for watch\_vols utility), etc.

**pool** A collection of tape cartridges having one or more similar features or attributes, such as a pool of scratch tapes.

**POST** Power-on self-test.

**priority CAP (PCAP)** A single-cartridge CAP used for priority entry and ejection of cartridges.

**processing errors** Errors that result from processing or network communication failures.

**PROM** Programmable read-only memory.

**PTP** See Pass-Thru Port.

## R

**RDBMS** Relational database management system.

**redo log files** Backup files used to restore the ACSLS database.

**relational database** A database that is organized and accessed according to relationships between the data items; relationships are represented by tables.

**ROM** Read-only memory.

**RPC** Remote Procedure Call.

## S

**SCAP** See standard CAP.

**scratch** An attribute of a tape cartridges, indicating that it is blank or contains no useful data.

**SCSI** Small computer serial interface.

**second disk journaling** Allows for the database's journal records to be written to a second disk device, instead of writing records to the primary disk. This improves the chances of recovery from a disk failure.

**Serial-attached** See HLI-attached.

**server system** The part of the library that is the residence for ACSLS, now referred to as

the Library Control System. The Library Control System acts as an interface between a library and client systems.

**server system user** A person who invokes ACSLS commands, utilities, or procedures on the server system. Server system users are generally site and maintenance personnel (for example, library operators, tape librarians, system administrators, CSEs, and systems personnel).

**servo** A system that uses feedback to control a process.

**silo** A commonly used term for an LSM. See Library Storage Module.

**SIMM** Single inline memory module.

**SQL** See structured query language.

**SRN** See service request number.

**SSI** See Storage Server Interface.

**SSR** Software Support Representative.

**Standard CAP (SCAP)** A 21-cartridge CAP with the storage cells arranged in three rows of seven fixed cells.

**Storage Server Interface (SSI)** A software component, resident on a client system, that translates and routes messages between client applications and the CSI.

**structured query language (SQL)** A language used to define, access, and update data in a database.

**StreamLine Library Console** the operator panel software application used for the StreamLine libraries.

**system resource variable** Used to control the amount of system resources used by ACSLS.

**system unit** The Library Control Platform.

## T

### **tape library management system (TLMS)**

A type of client application.

**TCP** Transmission Control Protocol.

**TLMS** See tape library management system.

**TOD** Time of day.

## U

**UDP** User Datagram Protocol.

**UNIX** An operating system originally developed by Bell Laboratories (now UNIX Systems Laboratories, Inc.) and used by a variety of computer systems.

**unsolicited messages** Messages that indicate an error or notify you when a particular routine action can be taken.

**UOC** Usable on codes.

**upgrade installation** Performed when installing a new version of ACSLS at an existing customer site.

**user** selectable features and options variables-Used to define various user-selectable features and options.

## V

**validation errors** Errors that result from format and syntax validation performed by cmd\_proc.

**venter** Virtual enter. Entering an unlabeled cartridge with a virtual label.

**virtual label** A logical volume ID (volser) that can be assigned to a cartridge when its physical label is missing or unreadable.

**volser** Volume Serial Number.

**volume** A data or cleaning cartridge.

**volume access control** Limits access to volumes, usually by the client.

**volume identifier** A six-character string that uniquely identifies a data or cleaning cartridge to the database.

**volume serial number (volser)** A synonym for external label identifier.

## W

**WTM** write tape mark.

## X

**XDR** External data representation.

**XML** Extensible Markup Language. A universal format for structured documents and/or data on the Web.



# Index

---

## A

- absent cartridge, 182
- absent volumes, 184
- access control
  - creating users, 143
  - enabling, 151
    - ownership.assignments file, 148, 163
  - how it works, 135
  - logging, 151, 158
  - maintaining, 158
  - modifying, 158
  - quick reference
    - for command access, 142
    - for volume access, 146
  - scenarios, 152
  - setting volume ownership, 147
- access control information, rebuilding, 129
- ACS states, 339
- ACSLS
  - audit, 163
  - commands, 289
  - database recovery, 220
  - de-installing, 47, 69, 96, 112
  - directory structure, 13
  - dual-LAN client configuration, 177
  - dual-LMU configurations, 175
  - error recovery, 503
  - event log, 505
  - extended store feature, 167
  - idling, 12
  - importing the database
    - Solaris 9, 109
  - installation script and utility logs, 508
  - installing
    - on Solaris 9, 101, 103
  - installing second disk support, 419
  - manually killing the ACSLS process, 7
  - mixed media and 3490E cartridges, 170
  - mixed-media restrictions for SCSI-attached LSMs, 171
  - mixed-media support, 169
  - overview, 1
  - reconfiguring
    - general product behavior, 121
    - logging options, 120, 127, 128
  - removing backup files, 38, 60, 104
  - removing previous versions, 38, 59
  - restarting, 12
  - restarting after installing second disk, 420
  - scratch preferences, 172
  - SCSI-attached libraries, 42, 63, 92
  - setting up client communication, 47, 68, 96
  - space required on second disk, Solaris 8/9, 413
  - starting, 3
  - stopping, 3
  - suspending request processing, 6
  - uninstalling, 47, 69, 96, 112
  - upgrading, 27, 28, 49, 50, 75, 76
  - using pkgadd, 40, 62
  - utilities, 241
  - verifying installation, 46, 67, 95
- ACSLS and OS diagnostic utilities
  - ck\_port\_node.sh, 509
  - probescsi.sh, 510
  - testlmu, 511
  - testports, 510
- ACSLS commands
  - audit, 294
  - cancel, 300
  - clear lock, 304
  - define pool, 306
  - delete pool, 308
  - dismount, 310
  - eject, 314

- enter, 318
- idle, 322
- lock, 324
- logoff, 326
- mount, 327
- mount \*, 330
- move, 335
- query, 337
- query acs, 338
- query cap, 340
- query clean, 342
- query drive, 343
- query lmu, 345
- query lock, 347
- query lsm, 349
- query mount, 350
- query mount \*, 353
- query pool, 355
- query port, 357
- query request, 358
- query scratch, 359
- query server, 361
- query volume, 362
- set cap mode, 365
- set cap priority, 367
- set clean, 369
- set commands, 364
- set lock, 370
- set owner, 372
- set scratch, 372
- show, 375
- start, 376
- switch lmu, 377
- unlock, 380
- vary, 382
- venter, 388

ACSL diagnostic information, gathering, 518

ACSL utilities

- bdb.acsss, 243
- cmd\_proc\_shell, 245
- config drives, 249
- db\_command, 252
- db\_export.sh, 253
- db\_import.sh, 254
- del\_vol, 254
- drives\_media.sh, 256
- ejecting.sh, 260
- find\_tty.sh, 261
- get\_license\_info, 265
- greplog, 266
- kill.acsss, 267
- rc.acsss, 274
- rdb.acsss, 275
- stats\_report, 280
- volrpt, 281
- watch\_vols, 286

acssa

- remote login, 8
- server login, 7

acssa user ID, 2

acsss, 555

acsss user ID, 2

acsss\_config, 14, 18, 555

- defining port connections, 510
- enabling or disabling auto-clean, 199
- for dynamic changes to drive configurations, 249
- logging volume statistics report, 236
- managing event log, 506
- menu, 117
- rebuilding access control information, 161
- run after adding, removing, or swapping modules, 501
- stats report, 281
- use to turn on firewall-secure option, 535
- used to enable access control, 151
- verifying database has been imported, 428

adding scratch volumes, 196

addresses, translating, 449

AIX

- configuring second disk, 422
- creating second disk file systems, 423

AIX installation from backup, 553

AIX system utilities

- diag, 512
- lsdev, 511

architecture, CSCI, 555

archived file, recovery, 229

audit command, 294

- audit, when to perform, 163, 229
- auditing the library, 163
  - audit intervals, 164
  - running an audit, 163, 229
- automatic database backup, 215

## B

- backup
  - ACSLs, 218
  - automatic, 215
  - default tape device attached to ACSLS server, 216
  - manual to tape, 216
  - specified tape device attached to ACSLS server, 217
  - to another server, 218
  - UNIX file, 218
- Backup and Recovery Tools, 435
  - automatic backups, 435
  - disaster recovery, 440
  - disaster scenarios
    - database corrupted-need to restore database to working state, 441
    - disk failure on primary disk with / export/home and second disk installed, 442
    - disk failure on second\_disk and second disk is installed, 443
    - primary disk failure and no second disk installed, 441
    - ran acsss\_config against wrong library, 441
    - server failure - rebuilding another ACSLS server with new hardware, 444
    - server failure - rebuilding the same server with new hardware, 443
  - importing database only from either a different platform or earlier release, 440
  - importing miscellaneous config files exported from different platform or earlier release, 440
  - manual backups, 436
  - manual database exports, 437

- merging customized dynamic variables from previous installation, 441
- restoring a backup created on a different server, 439
- restoring ACSLS non-database files, 439
- restoring from a previous tape or network file backup, 439
- restoring from current local disk backup, 438
- restoring from previous local disk backup, 438
- using db\_import.sh, 439
- using rdb.acsss, 438
- bdb.acsss utility, 243

## C

- cancel command, 300
- CAP
  - cap\_state, 341
  - displaying information about, 191
  - modes, 189
  - priorities, 191
  - set mode, 365
  - set priority, 367
  - states, 188
  - types, 187
- cartridge
  - recovery, 184
  - status reporting, 183
- cartridge access ports
  - enter/eject cartridges, 456
- cartridges
  - cleaning, correcting problems, 202
  - entering, 192
  - loading into disabled LSM, 203
- cell, display, 398
- ck\_port\_node.sh utility, 509
- cleaning cartridges, 369
  - checking, 342
  - defining, 200
- cleaning tape path, manual mode, 201
- cleaning transports, 199
  - auto clean option, 122
  - manually, 201

- cleaning transports in SCSI-attached LSMs, 202
- clear lock command, 304
- clustering
  - cartridges, 456
- cmd\_proc
  - batch user interface functionality, 11
    - redirecting output, 11
    - using an input file, 11
  - exiting, 6
  - input file, 11
  - logoff, 326
  - output file, 11
  - redirecting inputs and outputs, 11
  - redirecting output, 12
  - shortcut keys, 10
  - using, 4
  - using an output file, 12
- command processor
  - batch user interface functionality, 11
    - redirecting output, 11
    - using an input file, 11
  - exiting, 6
  - input file, 11
  - output file, 11
  - redirecting inputs and outputs, 11
  - redirecting output, 12
  - shortcut keys, 10
  - using, 4
  - using an output file, 12
- command syntax, 290
- commands, ACSLS, 289
- config acs, 247
- config drives utility, 249
- config lsm, 250
- config ports, 251
- Configuration
  - configuring your library hardware, 130
  - defining event notification settings, 129
  - menu, 116
  - rebuilding access control information, 129
  - setting access control variables, 127
  - setting automatic backup variables, 128
  - setting CSI tuning variables, 117
  - setting event logging variables, 120
  - setting general product behavior variables, 121
- configuration
  - updating, 130
- configuring
  - tape drives
    - layouts, 451
- configuring AIX
  - preparation, 100
- configuring second disk
  - AIX, 422
  - Solaris 8/9, 413
- contacting Sun Microsystems StorageTek Support, 1
- correcting cleaning cartridge problems, 202
- CSCI
  - architecture, 555
  - environmental variables, 558
  - error messages, 557
  - overview, 555
- customer support, 1
- customer-initiated maintenance (CIM), 1

**D**

- database
  - automatic backup, 215
  - backing up and restoring to another server, 218
  - backup to default tape device, 216
  - backup to specified tape device, 217
  - backup to UNIX file, 218
  - deleting volumes, 254
  - exporting, 206
  - importing, 209, 413, 427
  - installing, 206
  - manual backup to tape, 216
  - recovering and restoring, 220
  - recovery, 220
  - restarting, 231
  - restoring to most recent backup, 222
  - restoring to specified date and time, 223
  - verifying, 214
  - verifying and library configuration, 214
- db\_command utility, 252
- db\_export.sh utility, 14, 206, 253

- db\_import.sh utility, 254
- define pool command, 306
- de-installing second disk
  - AIX, 432
  - Solaris 8/9, 421
- del\_vol utility, 254
- delete pool command, 308
- deleting scratch pools, 197
- diagnostic information, AIX diag utility, 512
- diagnostic utilities
  - AIX, 511
- diagnostic utilities, ACSLS and OS, 509
- dismount command, 310
- display cap command, 396
- display cap information, 188
- display cell command, 398
- display command references, 393
- display commands, 394
  - display cap, 396
  - display cell, 398
  - display drive, 399
  - display lock, 401
  - display lsm, 403
  - display panel, 405
  - display pool, 406
  - display port, 408
  - display volume, 409
- display drive command, 399
- display lock command, 401
- display lsm command, 403
- display panel command, 405
- display pool command, 406
- display port command, 408
- display volume command, 409
- displaying scratch information, 195
- drive
  - clear lock, 304
  - configuration, 249
  - display, 399
  - lock, 324, 347, 370
  - state, 343
  - status, 343
  - unlock, 380
- Dual TCP/IP Support on SL8500, 478
- dual-LAN client configuration, 177
- dual-LMU, 345
  - managing, 175
- dual-LMU configuration, 175

## Index

- dynamic configuration
  - config acs, 247
  - config drives, 249
  - config lsm, 250
  - config ports, 251
- dynamic configuration utilities, 245
- dynamic variables
  - setting access control, 127
  - setting automatic backup, 128
  - setting CSI, 117
  - setting event logging, 120
  - setting event notification, 129
  - setting general product behavior, 121
  - using acss\_config, 116

## E

- eject command, 314
- ejected cartridge, 182, 186
- ejected volumes, 186, 194
- enabling access control, ownership.assignments file, 148, 163
- enter
  - cartridges, 454, 456
  - optimization, 456
- enter command, 318
- enter command, manual vs automatic, 319
- entering cartridges, 192
- error recovery, 503
- event log, 120, 505
  - managing, 506
  - using, 505
- event log, search, using greplog, 205, 506
- event notification, 180
- exiting a cmd\_proc, 6
- exporting
  - database to disk file, 207
  - database to tape, 208
- exporting the database, 206
- extended store feature, 167

## F

- file systems
  - creating second disk, 418, 423
  - creating second disk backup, 424
  - mounting second disk, 424
  - mounting second disk backup, 425
- find\_tty.sh utility, 261

firewall security  
ACSLs, 527  
benefits, 529  
communications, 540  
communications concerns, 528  
overview, 527  
RPC concerns, 528  
scenarios, 531  
security concerns, 528  
turning off, 538  
turning on, 535

float  
optimization, 456  
frequently asked questions, 542

## G

greplog, 205, 506

## H

HLI-PRC, description of, 447

## I

idle command, 322  
idling ACSLS, 12  
importing  
database from disk file, 210  
db\_import.sh, 254  
importing the database, 209, 413, 427  
improving library performance, 521  
initiating cmd\_proc, 7  
initiation, command file, 3  
installation script and utility logs, 508  
installing Solaris, preparation  
Solaris 8, 135  
Solaris 9, 28, 50, 76  
internal addresses  
description, 447  
invoking access control, ownership.assignments file, 148, 163  
IP addresses, registering for primary and secondary LANs, 177

## K

keyboard shortcuts, cmd\_proc, 10  
kill.acsss utility, 267

## L

level 0 archive, automatic database backup, 215  
library connections  
troubleshooting, 509  
library error recovery, 503  
library hardware, 28, 50, 76, 101  
library performance, improving, 521  
library server utilities, general information, 241  
licence key  
format, 23  
obtaining emergency key, 22  
obtaining permanent key, 22  
operational procedures, 24  
types, 21  
using licencekey.sh, 21  
utility, 21  
validation, 24  
LMU, switch, 377  
loading a CAP, 187  
loading cartridges into a disabled LSM, 203  
lock  
clear, 304  
command, 324  
lock command, 324  
logging, access control, 151, 158  
logical logs, automatic database backup, 215  
logoff command, 326  
lsdev, 511  
LSM  
display, 403  
populating, 186  
query, 349

## M

managing a mixed-media library, 169

- managing scratch volumes, 195
- manual enter, 318
- manual volume delete utility, 181, 254
- manually killing the ACSLS process, 7
- mask, network, 29
- mchanger
  - creating, 166, 518
  - removing, 166, 517
- mchanger, troubleshooting, 517
- media management, 547, 549
- media optimization, 454, 456
- miscellaneous ACSLS files, recovery, 229
- missing cartridge, 183, 185
- mixed media
  - 3490E cartridges, 170
  - files, 170
    - scratch preferences, 172
  - library, managing, 169
  - media label, 389
  - mount, 332
  - restrictions for SCSI-attached LSMs, 171
  - scratch preferences, 172
- mount \* command, 330
- mount command, 327
- mounting scratch volumes, 197
- move, 335
- move command, 335
- move volume, 335
- multi-homed host, 178

## N

- network IP addresses, registering for primary and secondary LANs, 178

## O

- optimization guidelines
  - cartridges, 454, 456
  - media, 454, 456
  - tape cartridges, 454, 456
- OS diagnostic utilities, 509
- OSLAN protocol, 138

## P

- panel, row, column addressing, 447
- Partitioning, 489
  - CAP behavior, 494
  - changing partition IDs, 491
  - changing partitioned ACS to un-partitioned, 493
  - guidelines, 489
  - library, 491
  - lre-partitioning a library, 492
  - software requirements, 489
- pass-thru, 521
- pool
  - define, 306
  - delete, 308
  - display, 406
  - query, 355
- port verification, 408, 510
- primary disk failure, recovering from, 225
- primary disk failure, recovery, 225
- probeFibre.sh, 509
- probescsi.sh utility, 510
- probe-scsi-all, 513

## Q

- query acs command, 338
- query cap command, 340
- query clean command, 342
- query commands, 337
- query drive command, 343
- query lmu command, 345
- query lock command, 347
- query lsm command, 349
- query mount \* command, 353
- query mount command, 350
- query pool command, 355
- query port command, 357
- query request command, 358
- query scratch command, 359
- query server command, 361
- query volume command, 362

## R

- rc.acsss utility, 274
- rdb.acsss, 275
- rdb.acsss utility, 275
- rebalancing scratch pools, 196
- reconfiguring ACSLS
  - logging options, 120, 122, 128, 129
- recovering from
  - primary disk failure, 225
  - second disk failure, 226
  - specific archived file, 229
- recovering from archived file, 229
- recovery
  - from failed server, 227
  - from primary disk failure, 225
  - from second disk failure, 226
- redirecting cmd\_proc inputs and outputs, 11
- registering for event notification, 180
- reporting volume ownership, 158
- request, query, 358
- restarting ACSLS, 12
- restarting ACSLS after installing second disk, 420
- restarting the database, 231
- restore, 220
  - ACSLs to another server, 218
  - miscellaneous ACSLS files, 229
  - the backup to a different server, 219
- restoring database
  - to another server, 218
  - to most recent backup, 222
  - to specified date and time, 223
- restoring miscellaneous ACSLS files, 229

## S

- scratch
  - cartridge optimization, 456
- scratch pool
  - defining, 306
  - deleting, 308
  - displaying, 406
  - mount scratch, query, 350
  - rebalancing, 196
- scratch, query, 359

- SCSI connection, verifying, 516
- SCSI-attached
  - adding, 165
  - verifying, 510
- second disk
  - creating backup file systems, 424
  - creating file systems, 418, 423
  - deinstalling on Solaris 8/9, 421
  - mounting backup file systems, 425
  - mounting file systems, 424
  - partitioning, 414
- second disk failure, recovering from, 226
- second disk file systems
  - creating on AIX, 423
- second disk, de-install
  - on AIX, 432
  - on Solaris 8/9, 421
- serial connection, verifying, 514
- server
  - failure recovery, 227
  - hardware, 28, 50, 76
  - idle, 322
  - starting, 376
- server state, query, 361
- server utilities, general information, 241
- set cap mode command, 365
- set cap priority command, 367
- set clean command, 369
- set lock command, 370
- set owner command, 372
- set scratch command, 372
- setting event notification, 129
- show command, 375
- showDevs.sh, 510
- SL500, 497
  - ACSLs Connectivity, 497
  - ACSLs limitations, 499
  - address schemes, 497
  - auditing the library, 500
  - CAP behavior, 500
  - column numbers, 498
  - drive addresses, 498
  - library configurable settings and ACSLS, 499
  - Library Differences, 497
  - panels, 498

- row numbers, 498
  - SL8500
    - ACSLs HA, 462
    - adding new SL8500 to right, 469
    - adding new SL8500 to the left, 468
    - adding new SL8500s, 463
    - CAP behavior, 453
    - Dual TCP/IP, 478
    - Expansion
      - auditing, 465
    - inoperative HandBot, 461
    - internal addresses and ACSLS addresses, 447
    - managing cartridge locations, 456
    - minimizing elevator and PTP activity, 454
    - missing cartridges, 457
    - partitioning, 489
    - pass-rhtu-ports
      - adding new SL8500s, 467
    - pass-thru ports, 466
    - pass-thru-ports
      - merging ACSs, 471
      - removing PTPs, 474
      - splitting ACSs, 474
    - supporting tape workloads, 455
    - tape drive locations, 451
    - translating internal addresses, 449
    - updating ACSLS after adding HandBots, 477
    - vary offline, 457
    - verify SL8500 components are operational, 447
  - Solaris
    - backup and export database, 28, 50, 76, 206
  - Solaris 10, 49
  - Solaris 10 x86, 75
  - Solaris 9, 27
    - installing, 29, 54, 81
    - preparing for installation, 28, 50, 76
  - Solaris system utilities
    - probe-scsi-all, 513
  - start command, 376
  - starting ACSLS, 3
  - starting cmd\_proc, 7
  - static variables, 115
    - setting CSI, 117
    - setting event logging, 120
    - setting event notification, 129
    - setting general product behavior, 121
    - using acsss\_config, 116
  - stats\_report utility, 280
  - stopping ACSLS, 3
  - Sun
    - worldwide offices, 2
  - suspending and resuming a cmd\_proc, 6
  - suspending request processing, 12
  - switch lmu command, 377
  - system information required for installation
    - Solaris 9, 28, 50, 76
  - System Utilities
    - AIX, 511
    - Solaris, 513
- T**
- TCP/IP timeout, 177
  - testlmu utility, 511
  - testports utility, 510
  - translating addresses, 449
  - transports
    - cleaning, 199
    - cleaning for SCSI attached LSMs, 202
    - cleaning manually, 201
    - displaying, 399
  - troubleshooting, 503
    - communication failures, 503
    - database recovery, 220
    - hardware failures, 504
    - library connection, 509, 514, 516
    - library connections, 509
    - SCSI connection, 516
    - serial connection, 514
    - software failures, 504
  - tuning variables, 117
- U**
- Uninstalling ACSLS
    - AIX, 112

- Solaris 10, 69
- Solaris 10 X86, 96
- Solaris 9, 47
- UNIX file backup, 218
- unlock command, 380
- unscratching volumes, 198
- user, show, 375
- using a cmd\_proc, 4
- using the CAP, 187
- utilities
  - bdb.acsss, 243
  - cmd\_proc\_shell, 245
  - config acs, 247
  - config drives, 249
  - config lsm, 250
  - config port, 251
  - db\_command, 252
  - db\_export.sh, 253
  - db\_import.sh, 254
  - del\_vol, 254
  - drives\_media.sh, 256
  - ejecting.sh, 260
  - find\_tty.sh, 261
  - free\_cells.sh, 262
  - get\_license\_info, 265
  - greplog, 266
  - kill.acsss, 267
  - moving.sh, 267
  - probeFibre.sh, 274
  - rc.acsss, 274
  - rdb.acsss, 275
  - stats\_report, 280
  - volrpt, 281
  - watch\_vols, 286

## V

- vary command, 382
- venter command, 388
- verifying ACSLS, 46, 67, 95, 110, 430
- verifying imported database, 214, 428
  - dismounting a volume to verify, 214
  - mounting a volume to verify, 214
- verifying library connection, 509
- volrpt utility, 281
- volume attributes, 209

- volume reports, custom, 234
- volume statistics, 120
- volume statistics report
  - logging volume movement, 236
- volume, querying, 362
- volumes
  - absent, 182, 183
  - adding scratch, 196
  - delete, 181
  - display, 409
  - displaying scratch information, 195
  - ejecting, 194
  - managing scratch, 195
  - mounting scratch, 197
  - unscratching, 198

## W

- watch\_vols utility, 286

## Z

- , 478

