

Oracle® DIVArchive
Installation and Configuration Guide
Release 7.3
E64033-05

September 2016

Oracle DIVArchive Installation and Configuration Guide, Release 7.3

E64033-05

Copyright © 2015, 2016, Oracle and/or its affiliates. All rights reserved.

Primary Author: Lou Bonaventura

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

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

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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

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

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

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

Table of Contents

1	INTRODUCTION.....	1
1.1	DOCUMENT PURPOSE AND SCOPE.....	1
1.2	DOCUMENT CONVENTIONS.....	1
1.3	DEFINITIONS, ACRONYMS, AND SPECIAL TERMS.....	2
1.4	SYSTEM OVERVIEW.....	4
1.4.1	DIVArchive Port Utilization.....	4
1.4.2	DIVArchive 7.3 New and Updated Functionality.....	5
1.4.3	DIVArchive 7.3 Configuration File Changes.....	7
1.5	DIVARCHIVE MEDIA STORAGE FORMATS.....	8
1.5.1	AXF Format.....	8
1.5.2	Tape Groups.....	8
1.5.3	Disk Arrays.....	9
1.6	DIVARCHIVE SOFTWARE OVERVIEW.....	10
1.6.1	Downloading DIVArchive Software.....	10
1.6.2	Core DIVArchive Software Components.....	10
1.6.3	DIVArchive Manager.....	10
1.6.4	DIVArchive Client API.....	10
1.6.5	DIVArchive Actor.....	10
1.6.6	Complex Objects.....	11
1.6.6.1	Complex vs. Non-Complex Objects.....	11
1.6.6.2	Complex Objects and FTP.....	12
1.6.7	DIVArchive Database Storage.....	16
1.6.7.1	DIVArchive Oracle Database.....	16
1.6.7.2	DIVArchive Metadata Database.....	16
1.6.8	DIVArchive Control GUI.....	16
1.7	ADDITIONAL DIVARCHIVE SOFTWARE COMPONENTS.....	18
1.7.1	Robot Manager.....	18
1.7.2	VACP Service.....	18
1.7.3	Storage Plan Manager.....	18
1.7.4	SNMP Agent.....	18
1.7.5	Access Gateway.....	19
1.7.6	Drop Folder Manager (DFM).....	19
1.7.7	Transcoder Support.....	19
1.7.8	Avid DHM Support.....	19
1.7.9	Avid DET Support.....	19
1.7.10	Avid Archive Manager Interface.....	20
1.7.11	DIVprotect.....	20

1.7.12	<i>DIVAnet</i>	20
1.7.13	<i>Object Transfer Utility (OTU)</i>	20
1.8	<i>DIVARCHIVE UTILITIES</i>	21
1.8.1	<i>DIVArchive Configuration Utility</i>	21
1.8.2	<i>Robot Manager Utilities</i>	21
1.8.3	<i>DIVArchive Backup Service</i>	21
1.8.4	<i>Scandrive Utility</i>	22
1.8.5	<i>Tape Reading Utility</i>	22
1.8.6	<i>DIVAscript</i>	23
1.8.7	<i>Recover Damaged Tape Utility (RDTU)</i>	23
1.9	<i>DIVARCHIVE SOFTWARE COMPONENT RELATIONSHIPS</i>	24
1.10	<i>DIVARCHIVE SOFTWARE COMPONENT DISTRIBUTION</i>	25
2	<i>DIVARCHIVE CONFIGURATION CONCEPTS</i>	26
2.1	<i>MODULE CONFIGURATION FILES</i>	26
2.2	<i>DIVARCHIVE DATABASE</i>	26
2.2.1	<i>Metadata Database</i>	27
2.2.2	<i>Metadata Database Sizing</i>	27
2.3	<i>ENVIRONMENTAL VARIABLES</i>	29
2.4	<i>DIVARCHIVE CONFIGURATION UTILITY OVERVIEW</i>	31
2.5	<i>CONNECTING TO THE DIVARCHIVE DATABASE</i>	32
2.6	<i>DISCONNECTING FROM THE DIVARCHIVE DATABASE</i>	33
2.7	<i>FRAME BUTTONS</i>	34
2.8	<i>CONTROL GUI PASSWORDS</i>	35
2.8.1	<i>Setting the GUI Administrator Password</i>	35
2.8.2	<i>Setting the GUI Operator Password</i>	36
2.8.3	<i>Changing the Database Log Level</i>	37
2.9	<i>SYSTEM TAB</i>	38
2.9.1	<i>Actor Configuration in the Database</i>	41
2.10	<i>ROBOTS TAB</i>	44
2.11	<i>DISKS TAB</i>	45
2.12	<i>DRIVES TAB</i>	46
2.12.1	<i>Drive Edit Dialog Window</i>	47
2.12.2	<i>Actors-Drives Area</i>	48
2.13	<i>TAPES TAB</i>	50
2.14	<i>SETS, GROUPS AND MEDIA MAPPING TAB</i>	52
2.15	<i>DIVAPROTECT TAB</i>	56
2.15.1	<i>Configuration Area</i>	57
2.15.2	<i>Event Definitions Area</i>	57
2.15.3	<i>Metric Definitions Area</i>	58

2.15.4	<i>Default Events and Metrics Configuration</i>	60
2.15.5	<i>Sample Metrics Definition</i>	62
2.16	<i>MEDIA TAB</i>	64
2.17	<i>STORAGE PLANS TAB</i>	65
2.18	<i>SLOTS TAB</i>	69
2.19	<i>MANAGER SETTING TAB</i>	72
2.19.1	<i>Media Configuration</i>	72
2.19.2	<i>Metadata Database Configuration</i>	73
3	INITIAL DIVARCHIVE CONFIGURATION SETTINGS	74
3.1	<i>PRODUCTION SYSTEM AND SITES DEFINITIONS</i>	75
3.2	<i>SOURCES AND DESTINATIONS DEFINITIONS</i>	78
3.3	<i>SOURCES AND DESTINATIONS CONFIGURATION CONCEPTS</i>	80
3.3.1	<i>Source and Destination Configuration Limitations</i>	80
3.3.2	<i>Source Type</i>	80
3.3.3	<i>Connect Options</i>	82
3.3.4	<i>Root Path</i>	82
3.3.5	<i>Metasource (Option)</i>	83
3.4	<i>ARRAYS AND DISKS</i>	83
3.4.1	<i>Defining an Array</i>	83
3.4.2	<i>Defining Disks</i>	85
4	ROBOT MANAGER CONFIGURATION	87
4.1	<i>CONFIGURATION OVERVIEW</i>	88
4.2	<i>SCSI CONNECTED LIBRARIES</i>	89
4.2.1	<i>Fiber Channel HBA's and SCSI Persistent Binding</i>	89
4.2.2	<i>Determining the SCSI Library Connection</i>	91
4.3	<i>SONY ODA DRIVES</i>	93
4.3.1	<i>Configuration File Changes</i>	97
4.3.2	<i>Configuration Utility Settings</i>	99
4.3.3	<i>Control GUI Settings and Information</i>	100
4.3.4	<i>Additional Information</i>	101
4.4	<i>CONFIGURING DIRECT ATTACHED SCSI LIBRARIES</i>	102
4.4.1	<i>Common Settings for SCSI Based Libraries</i>	102
4.4.2	<i>Additional Settings for Breecehill Libraries</i>	104
4.4.3	<i>Additional Settings for Qualstar and ADIC (Return Media Id Enabled) Libraries</i>	105
4.4.4	<i>Additional Settings for DVD Jukeboxes (ASACA or Plasmon)</i>	106
4.4.4.1	<i>DVD Jukeboxes Limitations</i>	106
4.4.4.2	<i>RobotManager.conf Options</i>	106
4.4.4.3	<i>Creating or Updating the JukeboxDB Database</i>	107
4.5	<i>CONFIGURING ACSLS ATTACHED LIBRARIES</i>	108
4.5.1	<i>LibAttach Configuration</i>	108

4.5.2	Testing LibAttach Connectivity to ACSLS	109
4.5.3	Firewall Support Notes	109
4.5.4	RobotManager.conf Common Options	110
4.6	CONFIGURING SONY PETASERVE LIBRARIES	111
4.6.1	RobotManager.conf Common Options	112
4.7	CONFIGURING ADIC LIBRARIES WITH SDLC	114
4.7.1	RobotManager.conf Common Settings	114
4.8	CONFIGURING SIMULATED LIBRARIES (FOR DIVARCHIVE SIMULATORS).....	116
4.8.1	RobotManager.conf Common Settings	116
4.9	ROBOT MANAGER COMMAND OPTIONS	118
4.9.1	Installing/Removing the RobotManager Service(s).....	118
4.9.2	RobotManager Service Management Functions.....	120
4.10	TESTING THE ROBOT MANAGER LIBRARY INTERFACE	121
4.10.1	Starting the Robot Manager	121
4.10.2	Testing the Robot Manager Library Control	122
4.10.2.1	Robot Manager Client	123
4.10.2.2	Robot Manager GUI	124
4.11	ROBOT MANAGER SYSTEM LEVEL CONFIGURATION (DIVARCHIVE CONFIGURATION UTILITY).....	125
4.11.1	Declaring the Robot Manager(s).....	125
4.11.2	Robot Manager – ACS Association.....	126
4.12	DIVARCHIVE ROBOT MANAGER LOGGING.....	127
4.12.1	Logging Mechanism	127
4.12.2	SCSI Error Management.....	128
4.13	MEDIA AND DRIVE TYPES CONFIGURATION	129
4.13.1	Configuration Overview	129
4.13.2	SCSI_drive_types / ACSLS_drive_types	131
4.13.2.1	SCSI_tape_types / ACSLS_tape_types	132
4.13.2.2	ADIC_media_types	133
4.14	SYNCHRONIZING MEDIA TYPES WITH THE DIVARCHIVE DATABASE.....	133
4.15	SYNCHRONIZING DRIVE TYPES WITH THE DIVARCHIVE DATABASE	135
4.16	TAPE CAPACITY AND BLOCK SIZE DEFINITIONS	137
4.17	SYNCHRONIZING MEDIA AND DRIVE COMPATIBILITY WITH THE DIVARCHIVE DATABASE	139
4.18	SYNCHRONIZING THE LIBRARY DRIVE LIST WITH THE DIVARCHIVE DATABASE.....	140
4.19	IDENTIFYING DRIVE SERIAL NUMBERS MANUALLY	142
4.19.1	Mounting/Dismounting a Tape	143
4.20	SYNCHRONIZING THE LIBRARY TAPES WITH THE DIVARCHIVE DATABASE	146
4.21	CREATING TAPE GROUPS	148
4.22	CREATING TAPE SETS.....	150
4.23	MEDIA REMAPPING.....	151

5	DIVARCHIVE ACTOR CONFIGURATION	152
5.1	<i>FILE PATH LIMITATIONS</i>	152
5.2	<i>CONFIGURATION OVERVIEW</i>	153
5.3	<i>LOCAL DIVARCHIVE ACTOR CONFIGURATION (ACTOR.CONF)</i>	154
5.4	<i>PARTIAL FILE RESTORE CONFIGURATION</i>	154
5.5	<i>INSTALLING/REMOVING DIVARCHIVE ACTOR SERVICE(S)</i>	162
5.5.1	<i>Actor Service Management Functions</i>	164
5.6	<i>LAUNCHING THE DIVARCHIVE ACTOR(S)</i>	165
5.7	<i>LOGGING</i>	166
5.8	<i>DIVARCHIVE ACTOR DEFINITIONS (CONFIGURATION UTILITY)</i>	167
5.8.1	<i>DIVArchive Actor Declaration</i>	167
5.8.1.1	<i>Actor Advanced Settings Panel</i>	170
5.9	<i>ACTOR TO DISK DEFINITIONS</i>	173
5.10	<i>ACTOR TO DISK INTERFACES AND MOUNT POINTS</i>	175
5.10.1	<i>Local</i>	175
5.10.2	<i>Remote</i>	175
5.10.3	<i>BML</i>	176
5.10.4	<i>FTP</i>	176
5.10.5	<i>MetaSAN</i>	176
5.10.6	<i>Simulation</i>	176
5.11	<i>ACTOR TO DRIVE CONFIGURATION</i>	177
6	DIVARCHIVE MANAGER CONFIGURATION	179
6.1	<i>CONFIGURATION OVERVIEW</i>	179
6.2	<i>LOCAL DIVARCHIVE MANAGER CONFIGURATION (MANAGER.CONF)</i>	180
6.2.1	<i>New and Updated Parameters Added to manager.conf</i>	180
6.2.2	<i>Basic Manager Settings</i>	182
6.2.3	<i>Database Settings</i>	183
6.2.4	<i>Advanced Settings</i>	184
6.2.5	<i>Logging Settings</i>	192
6.3	<i>REQUEST PRIORITY CONFIGURATION FILE</i>	193
6.4	<i>DESTINATION RE-ROUTING (RESTORE_TRANSLATIONS.CONF)</i>	195
6.5	<i>DIVARCHIVE MANAGER CONTROL OPTIONS</i>	197
6.5.1	<i>Installing/Removing the DIVArchive Manager Service</i>	197
6.5.2	<i>DIVArchive Manager Service Management Functions</i>	198
6.6	<i>LOGGING</i>	200
6.7	<i>CHECK CONNECTIVITY</i>	202
6.7.1	<i>Remote Client to DIVArchive Manager Connection Check</i>	202
6.7.2	<i>DIVArchive Manager to Actors Connection Check</i>	203
6.7.3	<i>DIVArchive Manager to Robot Managers Connection Check</i>	203

7	MANAGER FAILOVER PROCEDURES.....	205
8	DIVARCHIVE CHECKSUM SUPPORT CONFIGURATION	206
8.1	GLOBAL CHECKSUM PARAMETERS.....	206
8.2	CHECKSUM CONFIGURATION FOR SOURCE/DESTINATION	209
8.3	CHECKSUM CONFIGURATION FOR ARRAYS AND DISKS.....	213
8.4	CHECKSUM CONFIGURATION FOR GROUPS.....	216
8.5	CHECKSUM CONFIGURATION FOR ACTORS	217
8.6	ORACLE SAMMA SOLO INTEGRATION CONFIGURATION	218
8.7	AXF AND TEXT GENUINE CHECKSUM MODES	221
8.7.1	TEXT Genuine Checksum Mode	221
8.7.2	AXF Genuine Checksum Mode.....	223
9	TRANSCODER INSTALLATION AND CONFIGURATION	224
9.1	UPGRADING FROM TELESTREAM VANTAGE VERSION 5.0 OR EARLIER.....	224
9.2	INSTALLING TELESTREAM VANTAGE	224
9.2.1	Telestream License Installation.....	224
9.3	CONFIGURING DIVARCHIVE AND TRANSCODERS.....	225
9.3.1	General Instructions	225
9.3.2	Vantage Specific Configuration.....	226
9.3.2.1	Create the Output Path in Vantage	227
9.3.2.2	Minimum Vantage Workflow Requirements	229
9.3.2.3	Vantage Complex Workflow	233
9.3.3	Setting up Transcoders	237
9.3.4	Setting up Source/Destinations.....	237
10	MOVIE2ME INTEGRATION.....	239
10.1	MINIMUM REQUIREMENTS	239
10.2	MOVIE2ME WORKFLOW	240
10.3	MOVIE2ME CONFIGURATION.....	242
APPENDIX A – DYNAMIC CONFIGURATION CHANGES		247
A1	INTRODUCTION.....	247
A2	DIVARCHIVE MANAGER CONFIGURATION UPDATES	247
A3	CONFIGURATION UTILITY CHANGES IN THE SYSTEMS TAB.....	250
A3.1	Production Systems Frame.....	250
A3.2	Sites Frame	250
A3.3	Sources and Destinations Frame.....	251
A3.4	Actors Frame	252
A3.5	Transcoders Frame	253
A4	CONFIGURATION UTILITY CHANGES IN THE ROBOTS TAB	254
A4.1	Robot Managers Frame	254
A4.2	Media Compatibility Frame	254

A4.3	Robot Managers – ACS Frame.....	254
A5	CONFIGURATION UTILITY CHANGES IN THE DISKS TAB.....	255
A5.1	Arrays Frame.....	255
A5.2	Disks Frame	255
A5.3	Actor-Disk Connections Frame	256
A6	CONFIGURATION UTILITY CHANGES IN THE DRIVES TAB.....	257
A6.1	Drives Frame.....	257
A6.2	Libraries Frame	257
A6.3	Drive Properties Frame	258
A6.4	Actor – Drives Frame	258
A7	CONFIGURATION UTILITY CHANGES IN THE TAPES TAB.....	258
A8	CONFIGURATION UTILITY CHANGES IN SETS, GROUPS AND MAPPING TAB.....	259
A9	CONFIGURATION UTILITY CHANGES IN THE DIVAPROTECT TAB.....	259
A10	CONFIGURATION UTILITY CHANGES IN THE STORAGE PLANS TAB.....	259
A11	CONFIGURATION UTILITY CHANGES IN THE SLOTS TAB.....	259
A12	EVENT FIELDS AND METRIC DEFINITIONS.....	260
APPENDIX B – DIVARCHIVE SOURCES AND DESTINATIONS GUIDE.....		272
B1	INTRODUCTION.....	272
B1.1	Objective	272
B1.2	Scope	272
B2	GENERAL ITEMS.....	272
B2.1	Introduction.....	272
B2.2	Files Path Root.....	272
B2.3	Root Path	273
B2.3.1	UNIX style paths.....	275
B2.3.2	Windows Style Paths.....	275
B2.4	Metasource.....	276
B2.5	Connect Options.....	276
B2.5.1	Quality of Service (qos=).....	276
B2.5.2	Source/Destination FTP User Login (-login).....	278
B2.5.3	Source/Destination CIFS User Login (-user).....	278
B2.5.4	Source/Destination Password (-pass).....	279
B2.5.5	Source/Destination Connect Port (-port)	279
B2.5.6	Delete Source Content after Archiving (-allow_delete_on_source)	280
B2.5.7	Archive/Restore File Renaming Rules (-arch_renaming, -rest_renaming).....	280
B2.5.7.1	Renaming Rules Processing	281
B2.5.8	Files to be Skipped on Restoring (-rest_ignoring)	284
B2.5.9	Archive Files in a Specific Order (-file_order).....	285
B2.5.10	Specifying the Transcode Format (-tr_archive_format, -tr_restore_format).....	287

B2.5.11	Specifying a Transcoder Name (-tr_names).....	288
B2.5.12	Restoring Metadata (-rest_metadata or -rm)	289
B2.5.13	Restricting the Number of Actor Retries (-num_actors_to_retry).....	289
B2.5.14	MSS Source/Destination in MXF mode (-mxf)	290
B2.5.15	FTP Socket Window Size (-socket_window_size).....	290
B2.5.16	FTP Socket Block Size (-socket_block_size).....	291
B2.5.17	FTP Passive Mode Transfers (-pasv)	292
B2.5.18	Restore in AXF Mode (-axf).....	292
B2.5.19	Specifying Connection Timeouts (-list_timeout, -transfer_timeout, -control_timeout).....	293
B3	AVID MSS (PROGRAM STREAM).....	294
B3.1	Introduction.....	294
B3.2	Source/Destination Configuration	295
B3.2.1	MSS with Independent Storage	295
B3.2.2	MSS with Shared Storage.....	295
B3.2.3	MSS with Shared Storage (MXF mode).....	296
B3.3	DIVA_archiveObject Usage	296
B4	AVID AIRSPACE.....	297
B4.1	Introduction.....	297
B4.2	Source/Destination Configuration	297
B4.3	Source/Destination Usage	297
B5	AVID TRANSFER MANAGER (DHM INTERFACE).....	298
B5.1	Introduction.....	298
B5.2	Source/Destination Configuration	298
B5.3	Source/Destination Usage	299
B6	AVID TRANSFER MANAGER (DET INTERFACE).....	300
B6.1	Introduction.....	300
B6.2	Source/Destination Configuration	300
B6.3	Usage	300
B7	SEACHANGE BMS/BMC	301
B7.1	Introduction.....	301
B7.2	Source/Destination Configuration	301
B7.3	Source/Destination Usage	302
B8	SEACHANGE BML	303
B8.1	Introduction.....	303
B8.2	Source/Destination Configuration	303
B8.3	Source/Destination Usage	304
B9	SEACHANGE BMLE/BMLEX	305
B9.1	Introduction.....	305
B9.2	SAF Support.....	305

B9.3	Source/Destination Configuration	306
B9.4	Source/Destination Usage	307
B10	LEITCH VR SERIES	308
B10.1	Introduction.....	308
B10.2	Source/Destination Configuration	308
B10.3	Source/Destination Usage	308
B11	LEITCH NEXIO	309
B11.1	Introduction.....	309
B11.2	Source/Destination Configuration	309
B11.3	Source/Destination Usage	309
B12	GRASS VALLEY PROFILE.....	310
B12.1	Introduction.....	310
B12.2	Source/Destination Configuration	310
B12.3	Actor Configuration.....	310
B12.4	Network Configuration.....	311
B12.5	Source/Destination Usage	311
B13	GRASS VALLEY UIM.....	312
B13.1	Introduction.....	312
B13.2	Source/Destination Configuration	312
B13.3	Actor Usage.....	313
B14	GRASS VALLEY K2	314
B14.1	Introduction.....	314
B14.2	Source/Destination Configuration	314
B14.3	Actor Usage.....	315
B15	GRASS VALLEY M-SERIES IVDR	316
B15.1	Introduction.....	316
B15.2	Source/Destination Configuration	316
B15.3	Source/Destination Usage	316
B16	SONY MAV70.....	317
B16.1	Introduction.....	317
B16.2	Source/Destination Configuration	317
B16.3	Source/Destination Usage	317
B17	OMNEON SPECTRUM MEDIADIRECTOR (QUICKTIME)	318
B17.1	Introduction.....	318
B17.2	Source/Destination Configuration	319
B17.4	Source/Destination Usage	319
B18	OMNEON MEDIAGRID.....	321
B18.1	Introduction.....	321
B18.2	Source/Destination Configuration	321
B18.3	Source/Destination Usage	322

B19	QUANTEL POWER PORTAL	323
B19.1	Introduction.....	323
B19.2	Source/Destination Configuration	324
B19.3	Actor Configuration.....	324
B19.4	Source/Destination Usage	325
B20	SONY HYPER AGENT	326
B20.1	Introduction.....	326
B20.2	Source/Destination Configuration	327
B20.3	Source/Destination Usage	327
B21	STANDARD FTP/SFTP SERVERS	328
B21.1	Introduction.....	328
B21.2	Source/Destination Configuration	328
B21.3	Source/Destination Usage	329
B22	LOCAL	330
B22.1	Introduction.....	330
B22.2	Source/Destination Configuration	330
B22.3	Source/Destination Usage	330
B23	DISK / CIFS.....	332
B23.1	Introduction.....	332
B23.2	Source/Destination Configuration	332
B23.3	Source/Destination Usage	333
B24	METASOURCES	334
B24.1	Introduction.....	334
B24.2	Source/Destination Configuration	334
B24.3	Source/Destination Usage	337
B24.4	Archive Request with Delete on Source (-r and -delete_fpr).....	337
B25	EXPEDAT.....	340
B25.1	Introduction.....	340
B25.2	Source/Destination Configuration	340
B25.2.1	Enable Expedat Transfer Encryption (-encryption).....	341
B25.2.2	Expedat License Code (-license)	341
B25.3	Source /Destination Usage	342
APPENDIX C – ADIC SDLC INSTALLATION GUIDE		343
C1	SDLC SERVER	343
C1.1	Prerequisites	343
C1.2	Configuration	343
C2	SDLC CLIENT.....	345
C2.1	Prerequisites	345
C2.2	Installation	345
C2.3	Configuration	345

C3	TROUBLESHOOTING	347
C3.1	Useful <i>dasadmin</i> Commands	348
C3.1.1	Getting <i>dasadmin</i> help.....	348
C3.1.2	Mounting a tape.....	348
C3.1.3	Dismounting a tape	348
C3.1.4	Ejecting a Tape	348
C3.1.5	Inserting a Tape	348
C3.1.6	Querying Drives.....	348
C3.1.7	Retrieving the Tapes List	349
C3.1.8	Parking the Robot Arm	349
C3.1.9	Synchronizing the SDLC Database and Library	349
C3.1.10	Retrieving Tape Information.....	349
C3.1.11	Dasadmin Version Info	349
C3.1.12	Library Configuration Information.....	349
C3.1.13	Dasadmin References.....	349

Tables Index

Table 1:	Definitions, Acronyms, and Special Terms	2
Table 2:	DIVArchive Port Utilization	4
Table 3:	Software Component Distribution.....	25
Table 4:	Advanced Actor Parameters	41
Table 5:	Partial File Restore Parameters	42
Table 6:	Default DIVArchive Event and Metric Definitions.....	60
Table 7:	Description of Source Types	80
Table 8:	ContentDirector Attributes.....	82
Table 9:	robotmanager.conf Parameters	89
Table 10:	Common Settings for SCSI Based Libraries.....	102
Table 11:	SCSI Device Parameters	102
Table 12:	SCSI Module Parameters.....	103
Table 13:	Additional Settings for Breecehill Libraries	104
Table 14:	Additional Settings for Qualstar and ADIC Libraries	105
Table 15:	RobotManager.conf Options	106
Table 16:	RobotManager.conf Common Options	110
Table 17:	ACSLs Parameters.....	110
Table 18:	RobotManager.conf Common Options	112
Table 19:	SONY PetaSite Parameters	112
Table 20:	RobotManager.conf Common Settings.....	114
Table 21:	ADIC Parameters.....	114

Table 22: RobotManager.conf Common Settings.....	116
Table 23: Simulator Parameters	116
Table 24: Tape Drives and Associated Media	130
Table 25: Tape Capacities	137
Table 26: Tape Block Sizes	138
Table 27: Local DIVArchive Actor Configuration Options in actor.conf	154
Table 28: Partial Restore Parameters	155
Table 29: Maximum Operation Option Definitions	168
Table 30: Actor Advanced Parameters	171
Table 31: New manager.conf Parameters	180
Table 32: Basic Manager Settings	182
Table 33: Database Settings.....	183
Table 34: Advanced Settings	184
Table 35: Logging Settings	192
Table 36: Checksum Types and Associated ID Numbers	207
Table 37: Significance of Highlighted Fields in the Above Example	220
Table 38: Example Data in Reference to DFM Configuration Sample Above	220
Table 39: Root File Paths for Disks.....	222
Table 40: Root File Paths for FTP.....	222
Table 41: DIVArchive Manager Configuration Updates.....	247
Table 42: Production Systems Frame Parameter Changes	250
Table 43: Sites Frame Parameter Changes	250
Table 44: Sources and Destinations Frame Parameter Changes.....	251
Table 45: Actors Frame Parameter Changes.....	252
Table 46: Transcoders Frame Parameter Changes	253
Table 47: Robot Managers Frame Parameter Changes.....	254
Table 48: Media Compatibility Frame Parameter Changes.....	254
Table 49: Robot Managers – ACS Frame Parameter Changes.....	254
Table 50: Arrays Frame Parameter Changes.....	255
Table 51: Disks Frame Parameter Changes	255
Table 52: Actor-Disk Connections Frame Parameter Changes.....	256
Table 53: Drives Frame Parameter Changes.....	257
Table 54: Libraries Frame Parameter Changes	257
Table 55: Drive Properties Frame Parameter Changes.....	258
Table 56: Actor – Drives Frame Parameter Changes.....	258
Table 57: Configuration Utility Changes in the Tapes Tab	258
Table 58: Configuration Utility Changes in the DIVAprotect Tab	259
Table 59: Event Fields	260

Table 60: Metric Definitions	261
Table 61: Metrics Definitions.....	270
Table 62: Default Configuration	271
Table 63: Root Path Definitions	274
Table 64: UNIX Style Paths	275
Table 65: Windows Style Paths	275
Table 66: Quality of Service Connection Examples.....	277
Table 67: Renaming Rules Processing Examples.....	282
Table 68: Renaming Rules Processing Examples.....	283
Table 69: Renaming Rules Processing Examples.....	283
Table 70: MSS with Independent Storage.....	295
Table 71: MSS with Shared Storage	295
Table 72: MSS with Shared Storage (MXF mode)	296
Table 73: DIVA_archiveObject Source/Destination Usage Example	296
Table 74: Avid Airspace Source/Destination Attributes	297
Table 75: Source/Destination Usage Example	297
Table 76: Avid Transfer Manager (DHM interface) Source/Destination Attributes	298
Table 77: Avid Transfer Manager (DHM interface) Source/Destination Usage Example	299
Table 78: Avid Transfer Manager (DET Interface) Source/Destination Attributes	300
Table 79: Seachange BMS/BMC Source/Destination Attributes.....	302
Table 80: Seachange BMS/BMC Source/Destination Usage Example.....	302
Table 81: Seachange BML Source/Destination Attributes.....	303
Table 82: Seachange BML Source/Destination Usage.....	304
Table 83: Seachange BMLe/BMLex Source/Destination Attributes	306
Table 84: Seachange BMLe/BMLex Source/Destination Usage Example	307
Table 85: Leitch vR Series Source/Destination Attributes	308
Table 86: Leitch vR Series Source/Destination Usage Example	308
Table 87: Leitch Nexio Source/Destination Attributes	309
Table 88: Leitch Nexio Source/Destination Usage Example	309
Table 89: Grass Valley Profile Source/Destination Attributes	310
Table 90: Grass Valley Profile Actor Configuration Parameters	310
Table 91: Grass Valley Profile Source/Destination Usage Example	311
Table 92: Grass Valley UIM Source/Destination Attributes	312
Table 93: Grass Valley K2 Source/Destination Attributes.....	314
Table 94: Grass Valley M-Series iVDR Source/Destination Attributes.....	316
Table 95: Grass Valley M-Series iVDR Source/Destination Usage Example.....	316
Table 96: Sony MAV70 Source/Destination Attributes	317
Table 97: Sony MAV70 Source/Destination Usage Example	317

Table 98: Omneon Spectrum MediaDirector Source/Destination Attributes	319
Table 99: Omneon Spectrum MediaDirector Source/Destination Archive Parameters	319
Table 100: Omneon Spectrum MediaDirector Source/Destination Restore Parameters.....	320
Table 101: ContentDirector Attributes	321
Table 102: Omneon Mediagrid Source/Destination Usage Example	322
Table 103: Quantel Power Portal Sample System	323
Table 104: Quantel Power Portal Source/Destination Attributes	324
Table 105: Quantel Power Portal Actor Configuration Parameters.....	324
Table 106: Quantel Power Portal Source/Destination Archive Operation Usage.....	325
Table 107: Sony Hyper Agent Source/Destination Attributes	327
Table 108: Sony Hyper Agent Source/Destination Usage Example	327
Table 109: Standard FTP/SFTP Servers Source/Destination Attributes.....	328
Table 110: Standard FTP/SFTP Servers Source/Destination Usage Example.....	329
Table 111: Local Source/Destination Attributes.....	330
Table 112: Local Source/Destination Usage Example.....	330
Table 113: Disk/CIFS Source/Destination Attributes	332
Table 114: Disk / CIFS Source/Destination Usage Example	333
Table 115: Metasource Source/Destination Attributes.....	334
Table 116: Metasource Source/Destination Connection Options	336
Table 117: Source Destination Configuration.....	340
Table 118: SDLC Environment Variables.....	345
Table 119: Common Errors and Resolutions	347

Figures Index

Figure 1: Actor Connection Timeout during Complex Object Archive Via FTP	12
Figure 2: Socket Error during Complex Object Archive Via FTP	12
Figure 3: DIVArchive Configuration Utility Sources and Destinations Pane.....	13
Figure 4: Edit Source and Destinations Entry Window	13
Figure 5: Modify the Timeout Settings in the FileZilla Server Options	14
Figure 6: Windows Run Window	14
Figure 7: Windows Registry Editor	15
Figure 8: MaxUserPort Registry Entry	15
Figure 9: TcpTimedWaitDelay Registry Entry	15
Figure 10: DIVArchive 7.3 Control GUI Dashboard.....	17
Figure 11: GUI Dashboard Refresh Delay.....	17
Figure 12: Backup Service Timeout Setting	22

Figure 13: DIVArchive Software Component Relationships	24
Figure 14: Setting Windows Environment Variables.....	30
Figure 15: Connecting to the DIVArchive Database	32
Figure 16: Disconnecting from the DIVArchive Database.....	33
Figure 17: DIVArchive Frame Buttons.....	34
Figure 18: Setting the DIVArchive GUI Administrator Password	35
Figure 19: Setting the GUI Operator Password.....	36
Figure 20: Changing the DB Logging Levels.....	37
Figure 21: Remote Working Directory Sample Entry	38
Figure 22: Local Working Directory Sample Entry	39
Figure 23: Notifying Actor of Configuration Changes	39
Figure 24: DIVArchive System Tab Display Window	40
Figure 25: DIVArchive Robots Tab Window	44
Figure 26: DIVArchive Disks Tab Window.....	45
Figure 27: DIVArchive Drives Tab Window	46
Figure 28: Drive Edit Dialog	47
Figure 29: Drives Panel	47
Figure 30: Actors-Drives Area Close-up.....	48
Figure 31: Edit Actors-Drives Window.....	48
Figure 32: Edit Drives Entry Window.....	49
Figure 33: DIVArchive Tapes Tab Window	50
Figure 34: Edit Tape Properties Window.....	51
Figure 35: Add New Tape State Window	51
Figure 36: DIVArchive Sets, Groups & Media Tab Window.....	52
Figure 37: Edit Unused Tapes Sets Entry Window.....	53
Figure 38: Configuration Utility – Configuring the Format of a Tape Group	54
Figure 39: Add New Row in Media Mapping Window.....	54
Figure 40: Edit Media Mapping Entry Window	55
Figure 41: DIVAprotect Configuration Tab	56
Figure 42: DIVAprotect Tab Configuration Area Close-up.....	57
Figure 43: DIVAprotect Tab Event Definitions Area Close-up	57
Figure 44: Event Definition View Window.....	58
Figure 45: DIVAprotect Tab Metric Definitions Area Close-up.....	58
Figure 46: Metric Definition Editing Window	59
Figure 47: DIVAprotect Tab on the Configuration Utility and the “+” Button.....	62
Figure 48: Metric Definition Window.....	62
Figure 49: Media Tab Window	64
Figure 50: DIVArchive Storage Plans Tab Window	65

Figure 51: Add New Storage Plan Window	66
Figure 52: Edit Storage Plans Window.....	66
Figure 53: Add New Media Group Window	67
Figure 54: Edit Media Groups Entry Window	68
Figure 55: Edit Media Groups Warning Window.....	68
Figure 56: DIVArchive Slots Tab Window	69
Figure 57: Slot Configuration Window.....	70
Figure 58: Edit Slot Configuration Window.....	71
Figure 59: Configuration Utility Manager Settings Tab – Media Configuration.....	72
Figure 60: Configuration Utility Manager Settings Tab – Metadata Database	73
Figure 61: DIVArchive Configuration Overview	74
Figure 62: Production System and Sites Definitions.....	76
Figure 63: Example of Production Systems and Sites.....	77
Figure 64: Source/Destination Area in the Configuration Utility System Tab	78
Figure 65: Source and Destination Definitions	79
Figure 66: Connection Options	82
Figure 67: Disks Tab – Arrays Area Close-up	83
Figure 68: Add New Array Window	84
Figure 69: Edit Array Window	84
Figure 70: Editing an Array Warning Message.....	85
Figure 71: Disk Tab – Disks Area Close-up	85
Figure 72: Add New Disk Window.....	86
Figure 73: Edit an Existing Disk Window.....	86
Figure 74: Configuration Overview.....	88
Figure 75: Example of WWPN to SCSI ID Binding.....	90
Figure 76: Determining the SCSI Library Connection Using the Scandrive Utility.....	91
Figure 77: Windows Device Manager	92
Figure 78: Windows Device Manager Showing Sony ODA Drives	93
Figure 79: Optical Disc Archive Utility	94
Figure 80: Optical Disc Archive Utility Drive Settings Tab	95
Figure 81: Optical Media Information Screen.....	96
Figure 82: Configuration Utility Drives Tab.....	99
Figure 83: Configuration Utility Tapes Tab – Tape Properties Pane.....	99
Figure 84: DIVArchive Control GUI Tapes Window	100
Figure 85: Repack Window with Warning.....	100
Figure 86: Viewing a Mounted Drive in Windows Explorer	101
Figure 87: LibAttach Configuration.....	108
Figure 88: Installing/Removing the Robot Manager Service(s).....	119

Figure 89: Windows Services Applet and Registry Editor.....	120
Figure 90: Starting the Robot Manager in Windows	121
Figure 91: Windows Command Window	123
Figure 92: Robot Manager GUI.....	124
Figure 93: Declaring the Robot Manager(s)	125
Figure 94: Robot Manager – ACS Association.....	126
Figure 95: DIVArchive Robot Manager Logging Mechanism.....	127
Figure 96: Media and Drive Types Configuration Overview	129
Figure 97: <code>SCSI_drive_types</code> / <code>ACSLs_drive_types</code>	131
Figure 98: <code>SCSI_tape_types</code> / <code>ACSLs_tape_types</code>	132
Figure 99: <code>ADIC_media_types</code>	133
Figure 100: Synchronizing Media Types with the DIVArchive Database	134
Figure 101: Synchronizing Drive Types with the DIVArchive Database.....	136
Figure 102: Synchronizing Media and Drive Compatibility with the DIVArchive Database.....	139
Figure 103: Synchronizing the Library Drive List with the DIVArchive Database	141
Figure 104: Identifying Drive Serial Numbers Manually.....	142
Figure 105: Mounting/Dismounting a Tape	144
Figure 106: Running Scandrive Utility on Actor Host.....	145
Figure 107: Synchronizing the Library Tapes with the DIVArchive Database	147
Figure 108: Creating Tape Groups.....	148
Figure 109: Creating Tape Groups in DIVArchive	149
Figure 110: Creating Tape Sets	150
Figure 111: Media Remapping.....	151
Figure 112: Notifying Actor of Configuration Changes	153
Figure 113: Actor Configuration Overview.....	153
Figure 114: Configuration Utility Partial Restore Parameters	154
Figure 115: Actor Partial Restore Settings Tab	155
Figure 116: Installing/Removing DIVArchive Actor Service(s) in Windows.....	162
Figure 117: Windows Services Applet and Registry Editor.....	164
Figure 118: Launching the DIVArchive Actor(s) in Windows Service Manager.....	165
Figure 119: DIVArchive Actor Logs Window	166
Figure 120: DIVArchive Actor Declaration.....	167
Figure 121: Actor Settings Entry Screen	168
Figure 122: Actor Advanced Settings Panel.....	170
Figure 123: Actor Advanced Settings Tab.....	170
Figure 124: Actor to Disk Definitions.....	173
Figure 125: Actor-Disk Add New Connections Window.....	174
Figure 126: Actor-Disk Interfaces and Mount Points	175

Figure 127: Actor-Drive Configuration.....	177
Figure 128: Add New Actor-Drive Connections Window Showing Available Connections.....	177
Figure 129: Add New Actor-Drive Connections Window Showing No Additional Connections Available for This Actor.....	178
Figure 130: DIVArchive Manager Configuration Overview	179
Figure 131: Sample of <code>managerpriority.conf</code>	193
Figure 132: Destination Re-routing (<code>restore_translations.conf</code>)	195
Figure 133: Installing/Removing the DIVArchive Manager Service	197
Figure 134: DIVArchive Manager Service Management Functions	198
Figure 135: DIVArchive Manager Logging Window.....	201
Figure 136: Remote Client to DIVArchive Manager Connection Check.....	202
Figure 137: DIVArchive Manager to Actors Connection Check	203
Figure 138: DIVArchive Manager to Robot Managers Connection Check	204
Figure 139: Manager Tab of the Configuration Utility and Global Checksum Parameters	206
Figure 140: Manager: Checksum Feature Is Enabled	206
Figure 141: Manager: Default Checksum Type.....	207
Figure 142: Manager: Number of Retries Following Failed Checksum.....	208
Figure 143: Manager: Select Different Drive per Retry on Failed Checksum.....	208
Figure 144: Source/Destination Area in the Configuration Utility System Tab	209
Figure 145: Edit Sources and Destinations Popup Window	209
Figure 146: External Checksum Source Configuration.....	210
Figure 147: Selecting the Checksum Type Using the Pull-Down Box.....	210
Figure 148: Genuine Checksum Mode Pull-Down Box.....	211
Figure 149: Verify Following Archive Checkbox	211
Figure 150: Verify Following Restore Checkbox.....	212
Figure 151: Arrays and Disks Areas in the Configuration Utility System Tab.....	213
Figure 152: Verify Write Column in Arrays Area.....	213
Figure 153: Verify Write Column in Disks Area	214
Figure 154: Verify Write Edit Button for Specific Disks	214
Figure 155: Edit Row Popup Window.....	214
Figure 156: Verify Write Pull-down Box.....	215
Figure 157: Disk View in the Configuration Utility Showing Verify Write “On” For this Disk	215
Figure 158: Verify Write Column in the Groups Display	216
Figure 159: Actors Edit Row Popup Window Showing the Verify Tape Pull-down Box.....	217
Figure 160: Configuration Utility Transcoder Parameters.....	225
Figure 161: Vantage Transcode Limits	226
Figure 162: Edit Transcoders Entry Screen	227
Figure 163: Vantage Management Console Icon	227

Figure 164: Creating a New Variable	228
Figure 165: Create New Variable Window	228
Figure 166: Updating the Name and Saving the Variable.....	229
Figure 167: Vantage Workflow Designer Icon	229
Figure 168: Create New Category Window	229
Figure 169: Create New Workflow Window.....	230
Figure 170: Configuring the New Workflow - 1	230
Figure 171: Configuring the New Workflow - 2.....	231
Figure 172: Configuring the New Workflow - 3.....	231
Figure 173: Configuring the New Workflow - 4.....	231
Figure 174: Linking Receive and Flip Together.....	231
Figure 175: Flip Configuration Window	232
Figure 176: Receive Configuration Window	233
Figure 177: Releasing the Configuration to DIVArchive	233
Figure 178: Vantage Workflow Designer – Renaming the Workflow	234
Figure 179: Vantage Workflow Designer – Delete Watch.....	234
Figure 180: Vantage Workflow Designer – Replace Watch with Receive; Link to Identify	235
Figure 181: Deleting Deploy.....	235
Figure 182: Configuring Flip Factory.....	236
Figure 183: Final Complex Workflow Configuration	237
Figure 184: Configuration Utility Edit Source and Destinations Window - 1.....	238
Figure 185: Configuration Utility Edit Source and Destinations Window - 2.....	238
Figure 186: Movie2Me Basic Workflow	240
Figure 187: DIVArchive Array Entry Window.....	242
Figure 188: DIVArchive Disk Entry Window	243
Figure 189: DIVarchive Actor-Disk Connection Window.....	243
Figure 190: DIVarchive Sources and Destination Entry Window	244
Figure 191: Local Movie2Me Core Configuration	245
Figure 192: Remote Movie2Me Core Configuration	246
Figure 193: Skipping Files during Restore	285
Figure 194: Omneon Spectrum MediaDirector Connections	318
Figure 195: Quantel Power Portal Overview	323
Figure 196: SDLC GUI - Binding a Drive to a Client.....	344
Figure 197: Testing the SDLC Client/Server Connection	346

1 Introduction

1.1 Document Purpose and Scope

The Oracle DIVArchive Installation and Configuration Guide outlines the system in general, and initial configuration procedures for the Distributed Intelligent Versatile Archive (*DIVArchive*) platform from Oracle. This manual is written for Oracle delivery and support teams, resellers and integrators for initial system configuration, and end users for configuration changes (*only*).

The manual assumes a working knowledge of the Windows Operating Systems, and additional concepts such as networking, RAID, tape drive, and fiber channel technologies.

1.2 Document Conventions

The following conventions are used with respect to text:

Normal Standard Text.

Italic Used to emphasize a term or variable.

Bold Used to emphasize critical information.

6.1 Refers to a section or sub-section in the document.

`Courier New` Used for system screen output and system commands.

The following conventions are used with respect to file paths or variables:

`DIVA_HOME`: The Root Path on the file system where DIVArchive is installed.

The following conventions are used with respect to figures and drawings:

Red outlined boxes pointing to specific areas in a figure indicate procedural steps, or point out specific parameters being discussed in the section text.

Red outlined boxes that surround specific areas in a figure indicate specific areas of the figure being discussed in the section text.

1.3 Definitions, Acronyms, and Special Terms

Table 1: Definitions, Acronyms, and Special Terms

Term	Definition
Array	An Array designates a collection of disks designated by their name as they are declared in the DIVArchive Configuration. A Disk Name is associated with a mounting point. Archive Requests can be submitted with an array as the destination. DIVArchive is responsible for choosing the disk location to write the data when several disks belong to the same array.
AXF	The Archive eXchange Format (AXF) is based on a file and storage media encapsulation approach which abstracts the underlying file system, operating system, and storage technology making the format truly open and non-proprietary. AXF helps ensure long-term accessibility to valued assets, and keeps up with evolving storage technologies.
Category	Part of the access key to an Object. Categories are an approach to linking the object with the user activity field. It must not be confused with the Group idea, which is a storage concept.
Complex Object	An Object is defined as a Complex Object when it contains 1,000 or more components (<i>configurable</i>). Complex Object handling may differ from non-Complex Objects as noted throughout this document.
DPX	Digital Moving-Picture Exchange format. This is a high quality video format that consists of one or more files for each frame of video. This format is likely to be used with Complex Objects.
Jumbo Frames	Ethernet frames with more than 1500 bytes of payload. Conventionally, jumbo frames can carry up to 9000 bytes of payload.
Media Format	Tapes and Disks may be formatted as either AXF or Legacy (<i>format used prior to 7.0</i>). The format is set for Tape Groups and Disk Arrays during configuration.
MTU	Maximum Transmission Unit of a communications protocol of a layer is the size (<i>in bytes</i>) of the largest protocol data unit that the layer can pass onwards. A larger MTU brings greater efficiency because each packet carries more user data while protocol overheads, such as headers or underlying per-packet delays, remain fixed. A larger MTU also means processing of fewer packets for the same amount of data.

Term	Definition
Spanning	Splitting an Object's Component onto several tapes (<i>usually two</i>); this may occur when the component size is larger than the remaining size left on the initial tape.
WORM	Write-Once-Read-Many: A WORM disc can be written to only one time but can be read from many times. The WORM Media used in the Sony Optical Drives are Blu-ray Discs.
XML	Extensible Markup Language: a text-based database used to allow for the easy interchange of documents and data on the World Wide Web or between software components (<i>.xml</i>).

1.4 System Overview

Oracle DIVArchive 7.3 is capable of supporting interoperability among systems, helping to ensure long-term accessibility to valued content, and keeping up with evolving storage technologies. It offers profound present and future benefits for any enterprise that uses media – ranging from heritage institutions to schools, broadcasters, and simple IT-based operations.

Note: The File System Interface was not released with DIVArchive 7.x and is only supported by special request.

1.4.1 DIVArchive Port Utilization

The table below lists the standard ports used by the DIVArchive system. If you need assistance contact Oracle Services and speak with a Support Technician.

Table 2: DIVArchive Port Utilization

Service	Port Number	Description / Notes
FTP	21/tcp	Depending on configuration
Telnet	23/tcp	Solaris hosts only
SSH	22/tcp	Solaris hosts only
HTTP	80/tcp	DIVAdirector
Oracle SQLNet	1521/tcp	Manager database access
RDP - Microsoft Terminal Services	3389/tcp	Remote desktop
VNC	5900/tcp 5800/tcp	Used only when AMC is installed to access the AVID servers
DIVArchive RobotManager	8500/tcp	Robot Manager
DIVArchive Manager	9000/tcp	Manager
DIVArchive Actor	9900/tcp	Actor
DIVArchive FlipFactory	9000/tcp	FlipFactory
DIVArchive AMC	6101/tcp	AMC
DIVArchive Access Gateway	9500/tcp	DIVANet Access Getway

1.4.2 DIVArchive 7.3 New and Updated Functionality

The following is a listing of new and updated functionality beginning with DIVArchive version 7.3:

System:

- The DIVArchive 7.3 system uses the following 64-bit Oracle Java versions:
 - Java SE Runtime Environment build 1.8.0_45-b14
 - Java HotSpot Server VM build 25.45-b02, mixed mode
- System and option licensing is no longer required and has been removed.

Oracle DIVArchive Actor:

- Beginning with DIVArchive 7.3, the Actor can read instances with a format of AXF 0.9 and AXF 1.0, but can only write to the AXF 1.0 format. DIVArchive displays the version of AXF in use by an instance.
- A transcoder does not need to be coupled to a single actor. The actor column was removed from the Transcoders panel of the Configuration Utility. This also affects Flip Factory and Vantage configurations. The transcoder server and cache location is now embedded in the working directory as follows:

```
cifs://username:password@\\transcoder_ip\transcoder_folder
```

You may optionally specify a different location for the transcoder server than the location of the transcoder cache by specifying the following in brackets before the CIFS path:

```
transcoder:<user specified address>
```

You may also optionally link a specific actor to a transcoder by specifying the following in brackets before the transcoder specification that precedes the CIFS path:

```
actor:<user specified actor name>
```

- **Multiple transcoders are not supported for Flip Factory. They are only supported for Vantage.**
- **The original method of configuring a transcoder to a local Actor is still supported for legacy purposes.**
- The actor configuration settings have been moved from the actor configuration and Oracle Partial File Restore configuration files to the Configuration Utility with the exception of the Service Name and Port. These settings are located under Actor Advanced and Partial Restore Settings tabs of the Actor Pane of the Systems Tab. Some settings are only available in Engineering Mode.
- Transcoding now supports load balancing based on the following behavior:
 - Empty Queue
 - A transcoder not in use is selected first.
 - Performance
 - A higher performance transcoder will be selected next.
 - Number of free queue slots available.
 - A queue with more free slots will be selected next.
- Partial Restore support for QuickTime with MPEG2 LGOP (XDCAM HD 422/16 tracks of audio) has been added.

- If a transcoder address is not specified in the transcoder's working directory, a local transcoder address of 127.0.0.1 will be assumed as the transcoder address.
- IBM TS1150 drives are now supported.

Oracle Avid Connectivity:

- AMC has been tested and verified with Interplay 3.1.x and 3.2.
- TMC is not supported with Interplay releases 3.1.x, 3.2, and 3.5.

Oracle DIVArchive Manager:

- In the Control GUI, the media format for an instance will be Legacy, AXF 0.9, or AXF 1.0. Tapes, Groups, and Arrays configured as AXF will remain AXF as these media can hold AXF instances of version 0.9 **or** 1.0.
- GetObjectDetailsList (*GODL*) has been redesigned to be more efficient – there are no changes to API interaction for *GODL*.
- Two new error codes have been created for deleting objects:
 - `DIVA_ERR_OBJECT_BEING_ARCHIVED` is returned when you attempt to delete an object that is in the process of being archived.
 - `DIVA_ERR_OBJECT_IN_USE` is returned when you attempt to delete an object that is locked for reading, such as during (*but not limited to*) restore and copy commands.
- Some commands (*such as CopyToGroup and Delete*) did not previously have an **Options** field available. An **Options** field has been added to several commands in the 7.3 releases of the DIVArchive Java API and the DIVArchive C++ API.
 - Refer to the appropriate API documentation in the DIVArchive API Documentation library for detailed information.

Configuration Utility

- Prior to DIVArchive 7.3, the DIVArchive Media Mapping feature (*configurable from the Sets and Groups tab of the DIVArchive Configuration Utility*) only accepted input media values that existed in the DIVArchive database. The mapping feature has been enhanced to allow the mapping of any input string to media/storage plan values.

LocalDelete:

LocalDelete now has two modes of operation:

- The default legacy behavior called *ASAP Purge Mode*.
 - This mode purges objects immediately after they are copied to other sites.
- The new *Delayed Purge Mode*.
 - This mode operates with watermarks in a method similar to SPM. For more information please refer to the conf files LocalDelete configuration file folder.

Oracle DIVAnet

A new release of Oracle DIVAnet is now available beginning with DIVArchive release 7.3.1. Prior to this release, DIVAnet was installed, versioned, and released with the DIVArchive package. The new version of DIVAnet is named *Oracle DIVAnet v2*. It will be versioned independently of DIVArchive and has its own separate installation program.

Earlier DIVAnet versions are being renamed to *Oracle DIVAnet v1*. These versions were previously referred to as the *Access Gateway* and are still available.

DIVAnet v2 is configured differently from DIVAnet v1 and therefore is not a *drop-in* replacement. DIVAnet v2 requires that DIVArchive release 7.3.1 is installed on all sites.

Highlights of DIVAnet v2 include:

- New architecture supports more sites, requests, connections and objects.
- More configurable, manageable, flexible and has improved error handling.
- New DIVAnet User Interface (UI) has more powerful request monitoring, asset search capabilities, and a simple, more flexible reconciliation, copy and delete process.
- Enhance API functionality.

Refer to the *Oracle DIVAnet Guide* for more information on the new DIVAnet version.

Migration Service:

DIVArchive now includes an embedded migration service. It is a new and separate internal (*to DIVArchive*) service which helps users to schedule and run jobs to migrate content between different media inside of a DIVArchive system. You can use the Control GUI or command line client. Contact your Oracle Support Specialist for information about the embedded migration service.

Storage Plan Manager (SPM):

- A new parameter named `DELETE_OBJECT_ONLY_LAST_INSTANCE` works with the existing `ALLOW_OBJECT_DELETION` parameter. The difference between the two parameters is that `DELETE_OBJECT_ONLY_LAST_INSTANCE` will only delete an object if it is truly the last instance on the system. A complete description can be found in the SPM documentation and configuration file.
- SPM now supports disk cleaning based on the object's archived date. Previously, SPM's disk cleaning feature only supports cleaning based on an object's last access time and object size.

1.4.3 DIVArchive 7.3 Configuration File Changes

There have been changes made to the configuration files included with DIVArchive for version 7.3 that coincide with the new and changed functionality. This section gives a brief overview of the changes; more details may be found in the sections specific to each component.

- The Actor configuration settings have been moved from the actor and partial restore configuration files to the Configuration Utility.
 - The only remaining parameters in `actor.conf` are the Service Name and Port.
- The Flip Factory configuration file now allows users to specify a remote host using the `flipFactory.host` parameter added to the `ffsubmitter.properties.ini`.

1.5 DIVArchive Media Storage Formats

1.5.1 AXF Format

Archive eXchange Format (AXF) is an open format that supports interoperability among disparate content storage systems and ensures the content's long-term availability no matter how storage or file system technology evolves.

An AXF Object is an IT-centric file container that can encapsulate any number, and any type, of files in a fully self-contained and self-describing package. The encapsulated package contains its own internal file system, which shields data from the underlying operating system and storage technology. It's like a file system within a file that can store any type of data on any type of storage media.

Note: Complex Objects must be stored in AXF format. Because all Complex Objects are written in the AXF format, every instance of a Complex Object will be in the AXF format.

Tape Groups or Disk Arrays used by Complex Object requests must be in an AXF Format, as Complex Objects cannot be stored in Legacy Format. Because all Complex Objects are written in the AXF Format, any instance of a Complex Object will be in the AXF Format.

1.5.2 Tape Groups

In DIVArchive, a **Tape Group** or **Disk Array** has a **Media Format** parameter that indicates which **Storage Media Format** to use when creating new Archived Objects. The **Media Format** can be set to either DIVArchive **Legacy Format** or the **AXF Format**. This setting can be changed at any time and does not influence content already stored. This means that it is possible to have more than one Storage Media Format within Tape Groups and Disk Arrays.

A DIVArchive Object Instance is written in one and only one Media Format. Therefore, if an object spans tapes, each tape used as part of an object instance will be written in the same Media Format. An object can contain multiple instances, each of which can be stored in either **Legacy** or **AXF** format.

Although a Tape Group can contain more than one Storage Format, an individual tape has at most one Storage Media Format. The format of a tape instance is the format of the tape on which the instance resides. **All instances on a tape must have the same format.**

The Media Format for an empty tape is assigned when the first object on that tape is written. The tape is assigned the format of the Tape Group that appears in the request. After the Media Format for a tape is assigned, it cannot be changed unless all objects on the tape are deleted. Upon deletion of all objects from a tape, the tape's format becomes unassigned until content is again written to the tape.

Note: If the tape was in use, the tape format cannot change unless it is empty and cleared.

Both Legacy and AXF formatted tapes can exist in the same group. Nevertheless, objects in AXF Format will only be written to AXF formatted tapes, and objects in Legacy Format will only be written to Legacy formatted tapes even though they are in the same Tape Group.

Note: A Repack Request will always write the destination tape in the same Media Format as the source tape.

Similar to this, **Tape Spanning** operations will always use the same format across all tapes storing spanned objects. If an instance spans across multiple tapes, then all tapes used to span the content will have the same format.

1.5.3 Disk Arrays

Unlike tapes, disks do not have a format. DIVArchive allows storing Objects in different Media Formats on the same disk. If a disk contains objects in Legacy Format and that disk is then assigned to an AXF formatted array, it will still contain objects in Legacy Format. However, new objects written to the disk will be in AXF Format.

If a disk instance is non-complex and permanent (*not a cache instance*), it is stored in the format of the destination array. If a cache instance is non-complex, it is stored in the format of the group specified in the request.

To migrate objects from Legacy Media Format to AXF Media Format (*or back*), the **Copy To Group**, or **Copy As New** requests can be used. However some AXF Objects cannot be copied to the Legacy Format; copying objects from Legacy Format to AXF Format does not present any issues. In DIVArchive the only limitation on copying an Object Instance from AXF Format to Legacy Format is the Complex Object feature.

1.6 DIVArchive Software Overview

1.6.1 Downloading DIVArchive Software

You should stay current with the version of DIVArchive that you install and operate. Current versions of the software are found on the Oracle Software Delivery Cloud:

<https://edelivery.oracle.com>

Use the following procedure to obtain the software:

1. Log in to the system and search for “DIVArchive”.
2. Select the licenses you require (*for example, DIVArchive Actor, DIVArchive Manager, and so on*). You will need to search each time after adding a new license to the list.
3. For each license select the operating system you are running using the **Select Platform** button.
4. Continue through the download wizard, accepting the terms, until the final download page appears.
5. Confirm that all the licenses you require are listed.
6. Click the **Download All** button on the bottom right side of the screen, or click the file name link to download the software.

1.6.2 Core DIVArchive Software Components

The following modules are the core software components in a DIVArchive System.

1.6.3 DIVArchive Manager

The DIVArchive Manager (*Manager*) is the main component in a DIVArchive System. All archive operations are controlled and handled by the DIVArchive Manager. Operation requests are sent by initiator applications through the DIVArchive Client API. As a purchasable option, DIVArchive also supports Main and Backup DIVArchive Managers.

1.6.4 DIVArchive Client API

The DIVArchive Client API is a set of documented functions allowing external applications, acting as clients, to use the services offered by the DIVArchive System.

A library of client functions is provided and must be linked to each DIVArchive client application. These functions encapsulate client commands into DIVArchive request messages sent over a TCP/IP connection to the DIVArchive Manager.

Available APIs are C++, Java, and Web Services (WS). Refer to the appropriate API documentation in the DIVArchive API Documentation library for more information.

1.6.5 DIVArchive Actor

The DIVArchive Actor (*Actor*) is the data mover between devices in your production system. Actor supports interfacing and data transfer between many different types of devices.

All Actor operations are initiated and coordinated by the DIVArchive Manager via the TCP/IP networking protocol. Key benefits of the distributed design of the DIVArchive Actors are:

1. Additional DIVArchive Actors can be added to expand the archive sub-system to increase the overall bandwidth.
2. SAN based disk and tape drive resources can be shared amongst multiple DIVArchive Actors.
3. In combination with the DIVArchive Manager, multiple Actors provide scalability, load balancing, redundancy, and failover. Individual DIVArchive Actors can be seamlessly taken offline for maintenance without shutting down the DIVArchive System.

1.6.6 Complex Objects

Prior to DIVArchive 7.0, a limitation of 10,000 files per Object was enforced (*on some systems the limitation was adjusted to 15,000 files per Object*). With the introduction of the Complex Object feature, DIVArchive has significantly expanded these boundaries allowing up to 1,000,000 files and 10,000 folders. In addition, a Complex Object stores more information about the files and folders in an archive, such as subtotals for each directory. Finally, Complex Objects can serve as a platform to support more advanced GUI and API operations in addition to supporting future AXF features and functionality.

When an object is archived, DIVArchive determines whether the new object should be complex or non-complex based on its number of components (*files*). If the number of components is greater than 1,000 (*the default Complex Object threshold – see below on how to change this*), the object becomes a Complex Object; otherwise, the object is non-complex. After an object is deemed a Complex Object, it will always be complex – even if it is copied using the **Copy As** command, or imported using the Import/Export Utility.

Note: The `DeleteOnSource` option, `VFA/VFR` checksum verification workflows, and `getObjectListbyFileName/deleteFile` internal API calls are not supported by the **Complex Objects Workflows**.

1.6.6.1 Complex vs. Non-Complex Objects

A Complex Object differs from a non-Complex Object in some key ways. For instance, the file and folder metadata information of a Complex Object is stored in a file, not in the Oracle database. The file contains the filenames, folder names, checksums, and files sizes. The directory that contains these files is the Metadata Database Root Directory (*see below on how to configure this*). Complex Objects must be stored in AXF format – either on tape or on disk.

Because a Complex Object can contain hundreds of thousands of files, some DIVArchive API commands (*such as the `GetObjectInfo` command*) will not return the entire set of files. Instead, these commands return a single placeholder “file” which prevents downstream applications from being overwhelmed by file/folder information. A new DIVArchive API command has been created to return all of the files and folders within a Complex Object. Similarly, in the DIVArchive Control GUI, the entire set of files on a tape is not displayed in the **Object Properties** and **Tapes** dialogs – a single placeholder “file” is shown.

Not all DIVArchive operations are supported for Complex Objects. For instance, the *Delete on Source* feature is disabled for Complex Objects. The checksum features *Verify on Archive* and *Verify on Restore* are also disabled for Complex Objects. Oracle DIVAnet does not currently support replication of Complex Objects. Certain DIVArchive API operations used in Oracle DIVArchive AVID Connectivity (*such as `GetByFilename` and `DeleteByFilename`*) are not currently supported for Complex Objects.

A Complex Object maintains information about files in the archive as well as the folders. Complex Objects store subtotals for each folder, including the total number of files/subdirectories within the folder, as well as the total size of all files within the folder (*and within any subfolders*).

The *Complex Object Threshold* is a configurable parameter used by DIVArchive to determine whether a new object should be complex or not. If a new object has a number of components (*files*) that exceeds the threshold, the object automatically becomes a Complex Object. This value is set in the `manager.conf` configuration file. It is recommended that the threshold remains at the default value (*1,000 components*) unless there is a specific reason to adjust the value.

1.6.6.2 Complex Objects and FTP

When archiving Complex Objects via FTP protocol and using an FTP Client with default settings (*FileZilla is recommended*), the transfer will typically fail when archiving any object with more than approximately 3900 files. The reason for this is that occasionally, during the directory scan, the Actor connection times out before the size of the object can be computed as shown in the figure below.

Figure 1: Actor Connection Timeout during Complex Object Archive Via FTP

ID	Severity	Description	Date
930	Error	Request received : ABORTED	01/08/2014 11:13:14
929	Information	Request step is STEP_CLEANING_CACHE(actor2_70022, disk1_2)	01/08/2014 11:13:13
928	Information	Request step is STEP_WAITING_FOR_RESOURCES	01/08/2014 11:13:13
927	Error	The source/destination returned a command error: actor2_70022 returned:source/destination failure, list failed [421 Connection timed out.]	01/08/2014 11:13:13

More often, a request aborts in the middle of the transfer because the FTP Server (*FileZilla, etc.*) is consuming all of the available sockets.

Figure 2: Socket Error during Complex Object Archive Via FTP

ID	Severity	Description	Date
944	Error	Request received : ABORTED	01/08/2014 11:14:58
943	Information	Request step is STEP_CLEANING_CACHE(actor2_70022, disk1_2)	01/08/2014 11:14:57
942	Information	Request step is STEP_WAITING_FOR_RESOURCES	01/08/2014 11:14:57
941	Error	The source/destination returned a command error: actor2_70022 returned:source/destination failed to open file descriptor [425 Can't open data connection.]	01/08/2014 11:14:57

The first issue can be resolved by setting the following two parameters either in the **Source/Destination Command Options**, or in the options of the command itself:

- `-transfer_timeout 1200`
- `-list_timeout 600`

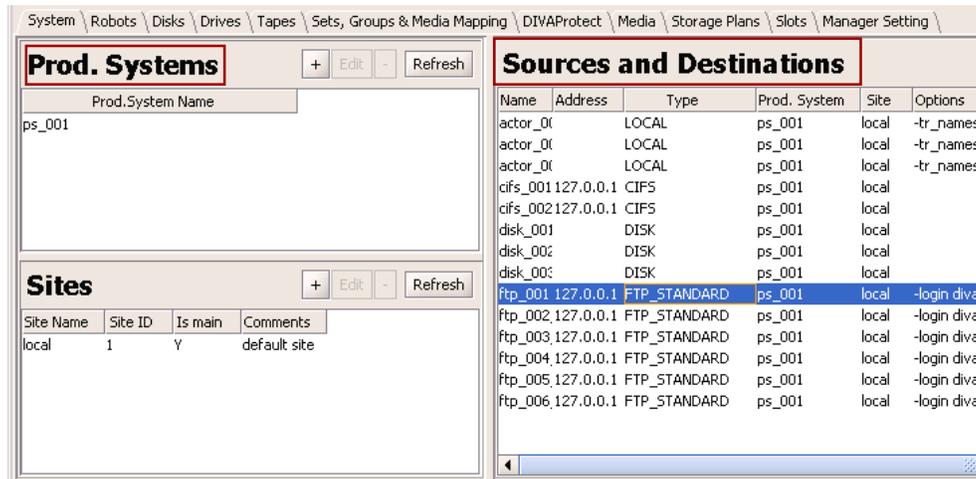
Note: For more information on these parameters, refer to **APPENDIX B – DIVArchive Sources and Destinations Guide**.

To include the parameters in the **Source/Destination**:

1. Start the **DIVArchive Configuration Utility**.

- In the **Sources and Destinations Pane** of the **System Tab**, open the **Edit Sources and Destinations Entry Window** by double-clicking on the **Source/Destination** you wish to modify.

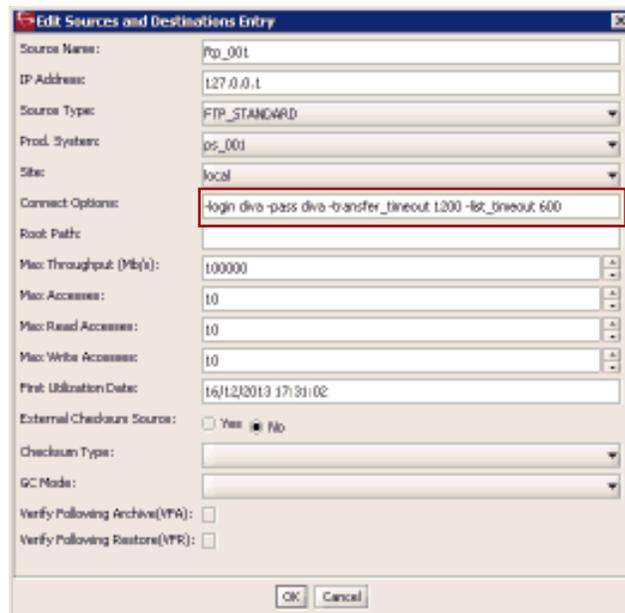
Figure 3: DIVArchive Configuration Utility Sources and Destinations Pane



- Add the following parameters to the **Connect Options** field:

`-transfer_timeout 1200 and -list_timeout 600`

Figure 4: Edit Source and Destinations Entry Window



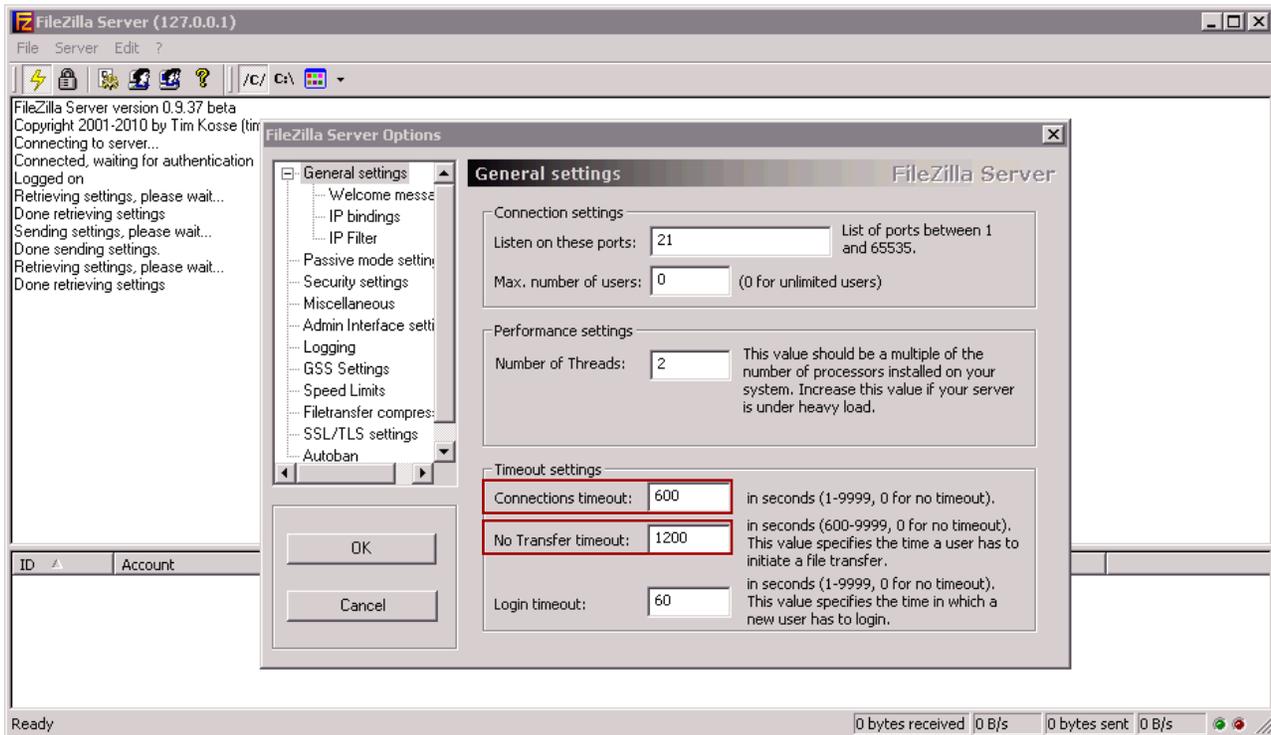
- Notify the Manager of the changes (**Control-N**) before trying to run the request in DIVArchive.

It is also recommended to set the corresponding parameters in the **FileZilla Server** under the General Settings:

- **Connections Timeout** = 600
- **No Transfer Timeout** = 1200

5. Open up the **FileZilla Server Interface** and select the Server Options (the 3rd icon on the toolbar) and modify under the **General Settings Window** as shown in the figure below.

Figure 5: Modify the Timeout Settings in the FileZilla Server Options

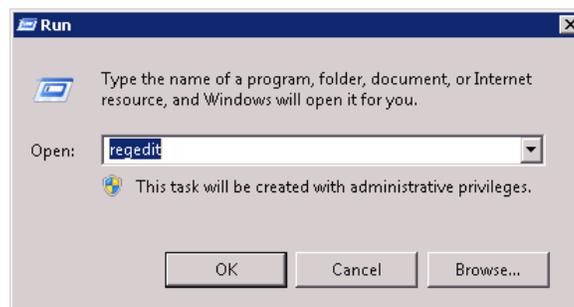


In the event of an abort, which may occur during transfer, there are two registry parameters that need to be created or modified (*typically created*):

- `TcpTimedWaitDelay = 10`
- `MaxUserPort = 90000`

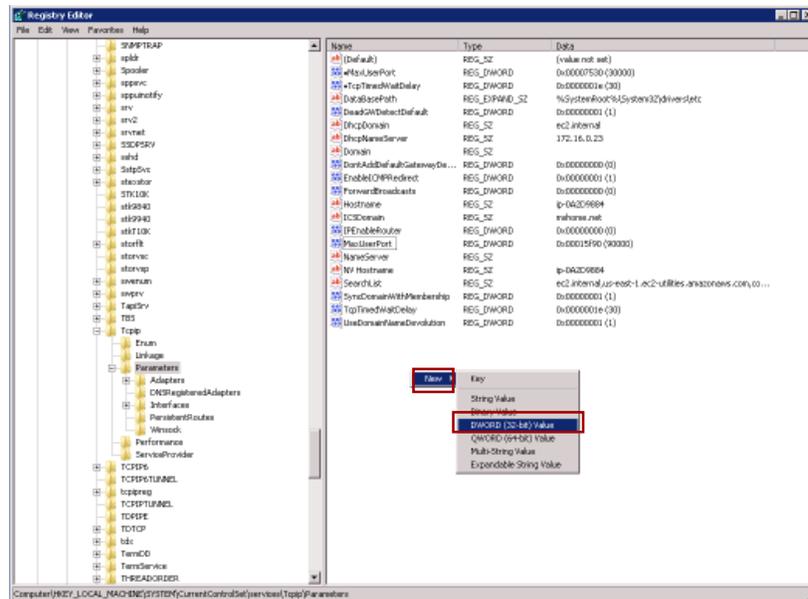
6. Click the Windows **Start Button**.
7. Click **Run**, and then type `regedit` in the **Run Window** text box.

Figure 6: Windows Run Window



8. Modify or create the following values under the `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters`:
 - `TcpTimedWaitDelay` = 10
 - `MaxUserPort` = 90000
9. If the desired registry parameter doesn't exist, right-click and create a new doubleword (*DWORD Value*) value:

Figure 7: Windows Registry Editor



10. If the parameter does exist, select it by double clicking on it and enter the values as shown below:

Figure 8: `MaxUserPort` Registry Entry

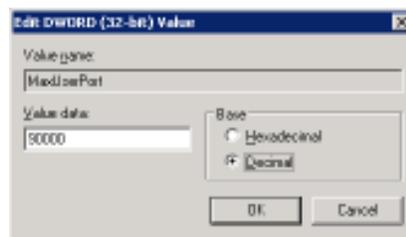
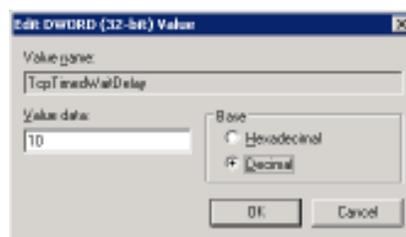


Figure 9: `TcpTimedWaitDelay` Registry Entry



11. Restart the machine to allow the new registry variables to take effect.

1.6.7 DIVArchive Database Storage

1.6.7.1 DIVArchive Oracle Database

The DIVArchive software is bundled with an Oracle database installation (*although Oracle v8 and v9 has been used in older versions of DIVArchive*). The database stores all information relating to the DIVArchive System, including its configuration. SQL queries used by the Manager are optimized to support configurations with up to 58 million components.

The Oracle database is not intended to be modified directly by customers, but rather by utilities from Oracle. *Direct modification of this database by customers through Oracle utilities is not supported by Oracle.*

Notes:

When installing DIVArchive in a 64-bit environment, the latest 64-bit DIVArchive Oracle version MUST be installed to utilize 64-bit support.

1.6.7.2 DIVArchive Metadata Database

To be able to effectively operate with large volumes of files/folders and other metadata, DIVArchive stores this metadata separately from the Oracle database in what is called the DIVArchive Metadata Database. The DIVArchive Metadata Database contains files stored in a file system local to the DIVArchive Manager. The directory that contains these files is the Metadata Database Root Folder.

The DIVArchive Metadata Database has a very high performance, almost unlimited scalability. The Metadata Database should be treated with the same caution as the Oracle database; it should be backed up at regular intervals via the DIVArchive Backup Service.

1.6.8 DIVArchive Control GUI

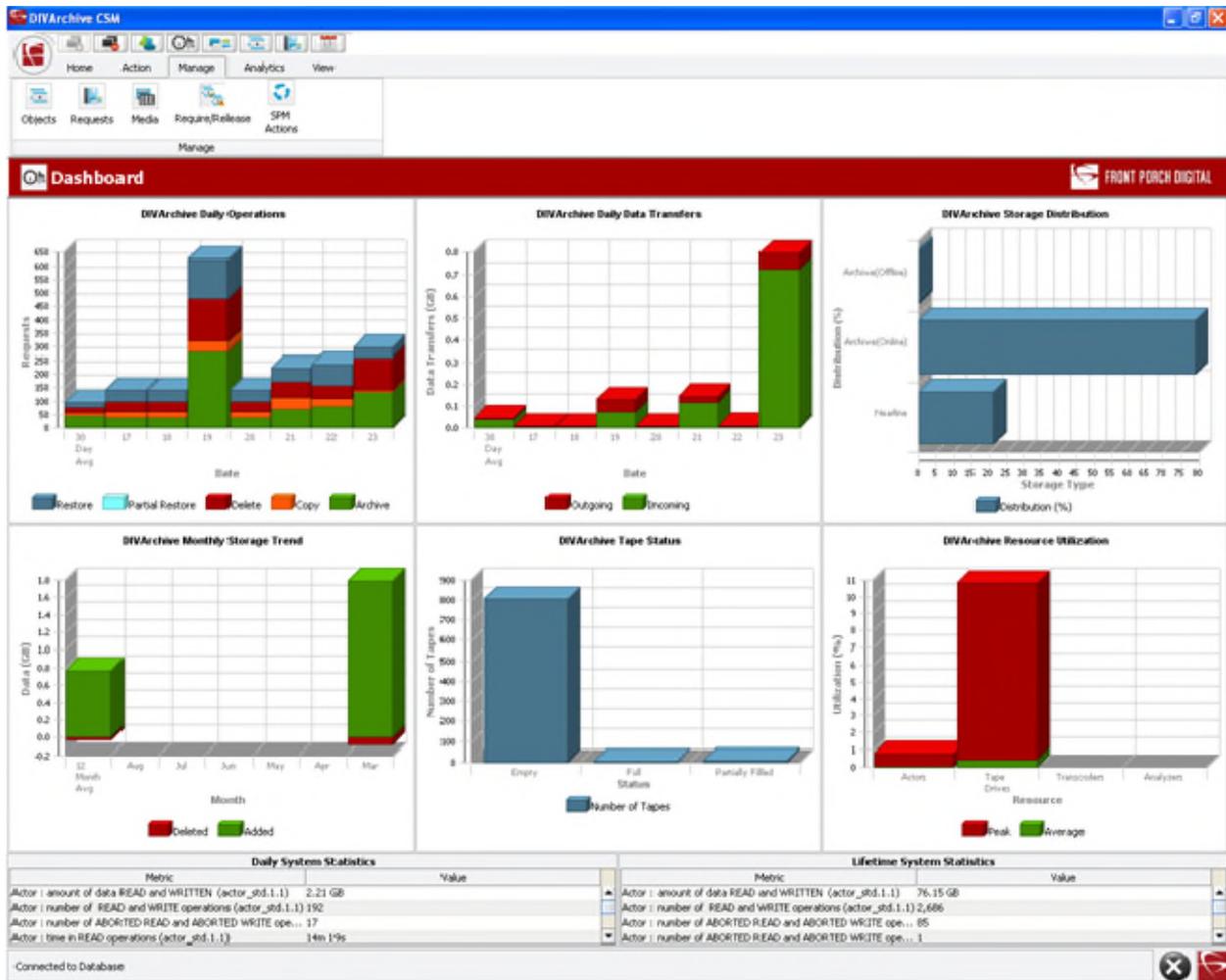
The DIVArchive Control GUI (*Graphical User Interface*) is a software utility that connects to both the DIVArchive Manager, and the DIVArchive Database, to monitor, control, and supervise operations in DIVArchive. Multiple DIVArchive Control GUI instances may be operated simultaneously from any machine that has TCP/IP connectivity to both the DIVArchive Manager and DIVArchive Database.

The DIVArchive Control GUI is not intended for the intensive archive operations of a DIVArchive System. Typically, archive operations are initiated to DIVArchive from a Broadcast Automation or MAM (*Media Asset Management*) system.

The DIVArchive Control GUI for DIVArchive version 7.3 offers an intuitive interface and provides for the additional functionality of the DIVArchive 7.3 system.

The Look-And-Feel and Navigation refreshes the user experience and delivers a modern and convenient way of navigating the DIVArchive GUI. The figure below represents the Control GUI Dashboard. Refer to the *Oracle DIVArchive Operations Guide* for more information on new features and using the new interface.

Figure 10: DIVArchive 7.3 Control GUI Dashboard



The refresh rate for the Control GUI is set in the Manager Setting Tab of the configuration Utility:

Figure 11: GUI Dashboard Refresh Delay

System \ Robots \ Disks \ Drives \ Tapes \ Sets, Groups & Media Mapping \ DIVAprotect \ Media

Manager Configuration

Parameter	Value
Media/Storage Plan Submission Delimiter	&
Maximum Number of Records in DP_OPERATIONS Table	500,000
Manager: Checksum feature is enabled	<input checked="" type="checkbox"/>
Manager: Default Checksum type	MD5
Manager: Number of retries following failed checksum	1
Manager: Select different drive per retry on failed checksum	<input type="checkbox"/>
GUI:Dashboard Refresh Delay	300,000
Manager: Set the default DIVArchive Backup Service monitor timeout(Minutes)	15
Complex Objects Metadata Database Location	
Enable/Disable Metadata Database Feature	<input type="checkbox"/>
Enable/Disable Database Backup Notification	<input type="checkbox"/>

1.7 Additional DIVArchive Software Components

Additional modules are available for purchase to expand the DIVArchive System further. Most of these options are currently covered in separate documents, but are briefly touched upon here for completeness.

1.7.1 Robot Manager

Although DIVArchive can only be used to manage disk storage, storage capacity can be further expanded by adding one or more tape libraries. In these cases, the DIVArchive Robot Manager Module provides an intermediate software layer for the DIVArchive Manager to interact with many different types of tape libraries. It is connected to the DIVArchive Manager via TCP/IP. This distributed architecture provides substantial flexibility:

1. Some libraries are controlled via a SCSI interface, which in turn is limited by its cable length. Since the connection to the DIVArchive Robot Manager from the DIVArchive Manager is over TCP/IP, the library does not need to be co-located near the DIVArchive Manager Host.
2. Enables installation of multiple and/or dissimilar libraries by configuring additional DIVArchive Robot Manager Modules.
3. Enables rapid development to support new types or models of libraries.
4. The robotics interface can be restarted without having to restart the DIVArchive Manager.

The DIVArchive Robot Manager interfaces with the library by using either a direct interface to the library itself (*via native SCSI or SCSI over Fiber Channel*), or via an intermediate Ethernet connection to the manufacturer's own library control software.

1.7.2 VACP Service

VACP (*Video Archive Communications Protocol*) is developed by Harris Automation Solutions and used by some automation systems for interfacing to an archive. DIVArchive has its own API for communicating with the DIVArchive Manager, which is not compatible with VACP.

To provide interoperability without the need to redevelop the archive interface at the automation level, this module is provided to act as an interface to convert VACP commands from the attached automation system, to DIVArchive API commands on hosts that have TCP/IP connectivity to DIVArchive.

1.7.3 Storage Plan Manager

The DIVArchive Storage Plan Manager (*SPM*) provides automatic migration and life cycling of material within the archive, based on the rules and policies defined in the SPM configuration. The DIVArchive Disk Space Monitor (*DSM*) works in conjunction with the SPM to delete material from SPM managed arrays (*based on disk space watermarks*).

1.7.4 SNMP Agent

The DIVArchive Simple Network Management Protocol (*SNMP*) Interface supports status and activity monitoring of different DIVArchive components. DIVArchive Management Information Base (*MIB*) is provided to third party SNMP monitoring applications.

The SNMP Agent utilizes the Windows SNMP Service.

1.7.5 Access Gateway

The DIVArchive Access Gateway provides DIVArchive client authentication and authorization. It can act as an intermediate gateway between DIVArchive components (*such as the VACP converter*) or third Party applications and the DIVArchive Manager, and can thus restrict that component's (*or application's*) access to the DIVArchive System.

Additionally, it is also used in DIVAnet installations and is the portal for multiple DIVArchive Systems to communicate with each other. Refer to the *Oracle DIVAnet Guide* for more information.

1.7.6 Drop Folder Manager (DFM)

The DIVArchive Drop Folder Manager (*DFM*) provides automatic monitoring of newly created files in a number of local directories or FTP folders (*or combinations thereof*). One file, or multiple files, per DIVArchive Object are supported. When a new file is identified, the DFM issues an archive request automatically to DIVArchive to archive the new file. After these files are successfully archived, they are then automatically deleted from the source. Refer to the *Oracle DFM User Guide* in the DIVArchive Additional Features library for more information.

1.7.7 Transcoder Support

The DIVArchive Actor can integrate with a transcoder engine to provide on-the-fly transcoding of material as it is archived or restored, or to create new objects from already existing content within the archive. Currently, integration to Bitscream products, Telestream's Flip Factory and Vantage are supported.

If a transcoder address is not specified in the transcoder's working directory, a local transcoder address of 127.0.0.1 will be assumed as the transcoder address.

Note: Multiple transcoders are not supported for Flip Factory. They are supported only for Vantage.

1.7.8 Avid DHM Support

The Avid Data Handler Module (*DHM*) interface support in DIVArchive allows finished content to be shared between post-production Avid environments and On-Air Video Servers. This eliminates the need for tape based content exchange. Time code based Partial File Restores of content to On-Air environments, and finished Avid Sequence submissions to On-Air servers are key to the DHM functionality offered within DIVArchive. DHM support is implemented in the DIVArchive Transfer Manager Communicator (*TMC*). Refer to the *Oracle Avid Connectivity User Guide* in the DIVArchive Additional Features library for more information.

1.7.9 Avid DET Support

The Avid Dynamically Extensible Transfer (*DET*) interface support in DIVArchive allows storage expansion of Avid Unity infrastructures and enables editors to move native Avid content in and out of the DIVArchive Storage System. Partially edited content stored within DIVArchive via the Avid DET interface can be later restored to Unity, and an editor can then resume editing where they left off. DIVArchive stores these files in native Avid format. DET support is implemented in DIVArchive Transfer Manager Communicator (*TMC*). Refer to the *Oracle Avid Connectivity User Guide* in the DIVArchive Additional Features library for more information.

1.7.10 Avid Archive Manager Interface

An interaction between the Avid Archive Manager Solution and DIVArchive is implemented in a separate service called the Archive Manager Communicator (AMC). AMC handles archive, restore, Partial File Restore, and delete commands from the Avid Archive Manager using DIVArchive to store Avid content in its native MXF OP1 Atom format. Refer to the *Oracle Avid Connectivity User Guide* in the DIVArchive Additional Features library for more information.

1.7.11 DIVAprotect

The DIVAprotect option is a utility that collects operational statistics from the DIVArchive System for the purpose of monitoring and maintenance of the archive's sub-components (*servers, media, drives, tapes, etc.*). Analysis of these statistics allows both proactive and reactive maintenance of the DIVArchive System. Refer to the *Oracle DIVAprotect User Guide* in the DIVArchive Additional Features library for more information.

1.7.12 DIVAnet

DIVAnet is a powerful feature that allows multiple DIVArchive platforms to exchange archive resources and/or content, whether the archive systems be local to each other or remote. Refer to the *Oracle DIVAnet Guide* for more information.

1.7.13 Object Transfer Utility (OTU)

The Object Transfer Utility (OTU) is an optional feature of the Control GUI, and provides a drag and drop interface to archive and restore material between DIVArchive and a (*supported*) source or destination server.

1.8 DIVArchive Utilities

1.8.1 DIVArchive Configuration Utility

The DIVArchive Configuration Utility is a software utility to configure the DIVArchive System. It connects directly to the DIVArchive Database and can be run on any machine that has TCP/IP connectivity to the host running the DIVArchive Database.

1.8.2 Robot Manager Utilities

During configuration and troubleshooting of the library and its tape drives, DIVArchive provides both a Command Line Interface (*CLI*) and Graphical User Interface (*GUI*) utility to send commands directly to the tape library via the Robot Manager.

These utilities are not (*and should not be*) used while the DIVArchive Manager is running as this can adversely affect archive operations.

1.8.3 DIVArchive Backup Service

To ensure reliability and monitoring of both the Oracle Database and Metadata Database backups, the DIVArchive Backup Service was introduced.

The DIVArchive Backup Service component is installed as an integral part of the standard DIVArchive System installation. The component is typically installed on the same server as the DIVArchive Manager and Oracle Database. The DIVArchive Backup Service allows for configuration of scheduled backups through its configuration file. The DIVArchive Backup Service manages and monitors the entire backup process.

When using Complex Objects, it is STRICTLY REQUIRED to use the Backup Service. The DIVArchive Backup Service is the only component backing up the Metadata Database and removing outdated Metadata files. When a Delete Request for a Complex Object is sent and processed, the data is removed from the Oracle Database, but the Metadata Database File is not deleted and is removed by the Backup Service after the configured clean up period (*define by the Recovery Period parameter*) has been reached. **Note: Do not change the Metadata Location parameter when the system is running.**

In the event of a database/system failure where restoring from a system backup is necessary, restoration of a stored backup is accomplished manually through existing Oracle scripts and should be performed by Oracle Support personnel only.

The DIVArchive Backup Service utilizes existing Oracle RMAN backup scripts to generate full database backups and incremental database backups. Oracle Database Backups and Metadata Database Backups will be incrementally replicated to all remote backup systems by the DIVArchive Backup Service.

DIVArchive Backup Service periodically sends status messages to the DIVArchive Manager. The DIVArchive Manager saves all error messages received in the Manager Events Log, and also forwards messages to all connected Control GUI applications to be displayed in a pop-up window. If no Control GUIs are connected at the time of the error, no error pop-ups will be displayed. However, errors can be reviewed later in the Events Log.

The service can be configured to monitor specific disks for space and send warnings and/or errors accordingly. Disks that can be monitored are C: and H: drives by default; however this configuration can be changed by modifying the `MONITORED_DRIVES=d1: ,d2:` parameter in the Backup Service Configuration File.

Parameter	Description
MONITORED_DRIVES=d1:,d2: (d1: and d2: represent the drive letters to be used)	Identifies the drive letters to be monitored by the DIVArchive Backup Service. The default is C: and H: and may be changed as required.

Users should have an elevated awareness of error messages from the Backup Service. Refer to the *Oracle DIVArchive Database User Guide* for more information.

Set the value for the Backup Service monitor timeout in the Manager Setting Tab of the Configuration Utility. The default setting is 15 minutes.

Figure 12: Backup Service Timeout Setting

The screenshot shows the 'Manager Configuration' window with a breadcrumb trail: System \ Robots \ Disks \ Drives \ Tapes \ Sets, Groups & Media Mapping \ DIVAprotect \ Media. The main title is 'Manager Configuration'. Below it is a table with two columns: 'Parameter' and 'Value'. The row 'Manager: Set the default DIVArchive Backup Service monitor timeout(Minutes)' is highlighted with a red border and has the value '15'.

Parameter	Value
Media/Storage Plan Submission Delimiter	&
Maximum Number of Records in DP_OPERATIONS Table	500,000
Manager: Checksum feature is enabled	<input checked="" type="checkbox"/>
Manager: Default Checksum type	MD5
Manager: Number of retries following failed checksum	1
Manager: Select different drive per retry on failed checksum	<input type="checkbox"/>
GUI:Dashboard Refresh Delay	300,000
Manager: Set the default DIVArchive Backup Service monitor timeout(Minutes)	15
Complex Objects Metadata Database Location	
Enable/Disable Metadata Database Feature	<input type="checkbox"/>
Enable/Disable Database Backup Notification	<input type="checkbox"/>

1.8.4 Scandrive Utility

This utility is provided on Windows platforms only to aid in obtaining detailed device information, such as serial numbers, firmware versions, and SCSI information from tape libraries or tape drives, for use in the DIVArchive configuration.

1.8.5 Tape Reading Utility

This utility is provided on Windows platforms and is primarily used in conjunction with the Robot Manager Client utilities to send manual Eject commands to a tape drive connected to an Actor Host. This utility also provides advanced tape based operations, such as tape formatting, but should only be used under guidance from Oracle Technical Support.

Note: This utility is not (and should not be) used while the DIVArchive Manager is running.

1.8.6 DIVAscript

This utility allows DIVArchive API commands to be executed via UNIX or DOS based scripts. It is designed to run automated tasks for testing rather than for any intensive uses.

1.8.7 Recover Damaged Tape Utility (RDTU)

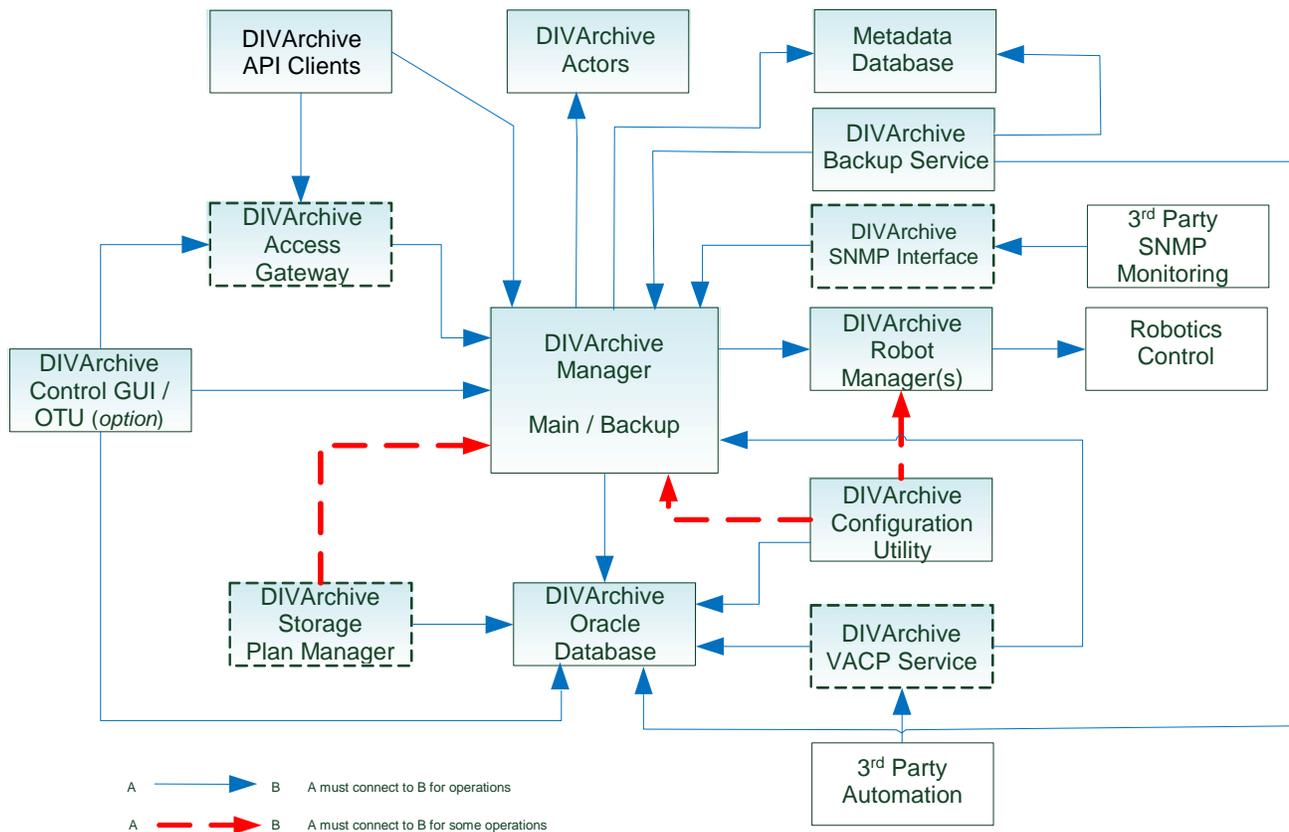
The Recover Damaged Tape Utility (*RDTU*) is designed to recover object instances that are contained on a damaged tape. The utility is able to recover instances that have valid copies on other available media (*i.e. internal tape or connected disk/array*) within a local or remote DIVArchive System.

1.9 DIVArchive Software Component Relationships

The following diagram shows the relationships and/or dependencies between the software components of a DIVArchive System. It specifically points out the client/server links between them. A client/server link can be interpreted as, “this client must connect to that server in order to be considered as operating”.

On the other hand, a client/server link between two components does not necessarily mean that the server software must be started before the client. An example is the DIVArchive Manager/Actor connection. Each Actor acts as a server and the Manager initiates a client connection to the Actor. An Actor however can be launched after the Manager is running since the Manager will attempt to reconnect to the Actor at periodic intervals.

Figure 13: DIVArchive Software Component Relationships



Note: DIVArchive can run independently of the Control GUI and/or Configuration Utility and they can be launched at any time after the DIVArchive Manager is running.

1.10 DIVArchive Software Component Distribution

The DIVArchive platform is flexible and scalable, so the installation of some software components can vary depending upon the degree of storage and servers that are managed. Small installations may have all DIVArchive software components installed on a single host, whereas a very large installation will have these components distributed amongst several servers. When run on Windows servers, all of these components run as system services.

Table 3: Software Component Distribution

Software Component	Typically Installed On:
DIVArchive Manager(s)	Main and Backup DIVArchive Manager Servers
DIVArchive Oracle Database	Main and Backup DIVArchive Manager Servers
DIVArchive Metadata Database	Main and Backup DIVArchive Manager Servers
DIVArchive Backup Service	Main and Backup DIVArchive Manager Servers, and Actors
DIVArchive Robot Manager(s)	Main and Backup DIVArchive Manager Servers <i>(Robot Manager can also be installed on a separate server when the tape library is installed a substantial distance from the DIVArchive Manager servers).</i>
DIVArchive Storage Plan Manager	Main and Backup DIVArchive Manager Servers
DIVArchive VACP Service(s)	Main and Backup DIVArchive Manager Servers
DIVArchive SNMP Agent	Main and Backup DIVArchive Manager Servers
DIVArchive Access Gateway	Main and Backup DIVArchive Manager Servers
DIVArchive Actor(s)	DIVArchive Actor Server(s)
DIVArchive TM Communicator	DIVArchive Actor Server(s)
DIVArchive Archive Manager Communicator	DIVArchive Actor Server(s)
DIVArchive Drop Folder Monitor (DFM)	DIVArchive Actor Server(s)

2 DIVArchive Configuration Concepts

2.1 Module Configuration Files

Each DIVArchive software module has its own static configuration text file with parameters needed to launch that particular application, typically denoted with the `.conf` extension. There are some DIVArchive modules that use an XML based file rather than a text file for their configuration and will be noted where applicable.

Unlike older versions of DIVArchive that had stored these configuration files in the same folder as the application itself, DIVArchive 7.3 centralizes them to a dedicated `conf` sub-folder under the DIVArchive Program Group.

These configuration files are typically updated with additional or changed settings in newer releases of the software. A new or patch version of DIVArchive will have the new versions of the `.conf` files appended with an `.ini` extension. For example, the new version of the DIVArchive Manager Configuration file will be named `manager.conf.ini`. After the installation is complete and the new configuration file updated, the `.ini` extension needs to be removed.

Each configuration file can be opened and edited with any plain text editor (*e.g. Notepad on Windows*).

Any changes made to the configuration file of a DIVArchive Software Component requires that the component itself needs to be shut down and then restarted for the changes to take effect. The exception are the Manager and Access Gateway options, which both allow configuration changes to be reloaded while they are still running. Be aware that there are co-dependencies between some applications in the DIVArchive platform, so other components may also need to be restarted as well for any changes to take effect.

2.2 DIVArchive Database

At the system level, settings that relate to the overall operation of each DIVArchive Component and their interaction are configured and retained by an Oracle Database. This is commonly known (*and will be referred to in this document*) as the DIVArchive Database (*or just simply as the database*).

User modification of this database is performed by the DIVArchive Configuration Utility.

The DIVArchive Configuration Utility connects only to the database and does not necessarily require the DIVArchive Manager to be running. It is only intended for experienced users and caution should be exercised when altering settings via this utility. An incorrect setting can impede DIVArchive operations or prevent the DIVArchive Manager from starting successfully. If you are unsure about making a particular change, you should contact Oracle Technical Support for assistance.

When launched, the DIVArchive Manager gathers the DIVArchive System Configuration from the database. However, it does not poll the database for changes made through the Configuration Utility and therefore the database must be notified of any changes made. This is performed by the **Notify Manager** command in the Configuration Utility.

Most changes to the configuration can be accomplished while the DIVArchive Manager is running. There are a small number of configuration changes that require a restart of the DIVArchive Manager to become effective. A full list of changes that can be made to the system configuration dynamically while the DIVArchive Manager is running is listed in Appendix A.

The Configuration Utility itself also does not dynamically poll the database for changes that are made to it when the Manager is running. In such cases, an **Update** button is provided in the utility to refresh the displayed information from the database.

The Configuration Utility can be installed on any machine that has TCP/IP connectivity to the database and a supported Java Runtime Environment installed. DIVArchive version 7.3 requires the Java Runtime Environment 64-bit (*build 1.8.0_45-b14*), to be installed in order to launch the Configuration Utility successfully.

In some cases, a network firewall between the two can prevent a connection. For complete operation and functionality of the Configuration Utility, the Oracle Listener Port (*typically 1521*) and the DIVArchive Robot Manager Ports (*typically 8500 and up*) need to be opened in the firewall. Full functionality of the Control GUI also requires that the DIVArchive Manager Port (*typically 9000*) is open.

2.2.1 Metadata Database

The DIVArchive Metadata Database has a very high performance, almost unlimited scalability. The Metadata Database should be treated with the same caution as the Oracle database; it should be backed up at regular intervals via the DIVArchive Backup Service.

It is highly recommended that the Metadata Database Files be stored on a RAID Disk Array. The Metadata Database should not be on a standard disk due to decreased performance and the (*real-time*) backup functionality that a RAID Array offers the system.

Metadata Database Files stored on a standard disk will be vulnerable to data loss in the event of a single disk failure, until the information is replicated via the DIVArchive Backup Service. Storing the Metadata Database Files on a RAID Array isolates the data from this type of failure.

Note: The information stored in the Oracle Database is already stored on a RAID-1 Array and is not subject to data loss in the event of a single disk failure.

Refer to 2.19.2 for Metadata Database configuration information.

2.2.2 Metadata Database Sizing

What is the **MINIMUM** amount of disk space is needed to support the Complex Object Metadata Database necessary? The following formula can be used as a rough guide:

$$(100 + \text{average_path_filename_size}) * 1.15 * \text{avg_num_component_files} * \text{num_objs}$$

Example:

- `average_path_filename_size = 60`
 - `this/nested/subdir01/As_The_World_Turns_24fps_scenes1-10.avi`
- `avg_num_component_files = 200,000`
 - The Average Number of Files and Folders within the Complex Object.

- `num_objs = 50,000`
 - The Number of Complex Objects to be archived.

In this example, **MINIMUM** budgeting for a Metadata Database size of around 1.67TB would be recommended.

When planning the system, Metadata Database disk space should be selected to ensure (un)expected growth of the environment.

Note: The same disk space must be allocated for the Metadata database in all of the backup systems.

2.3 *Environmental Variables*

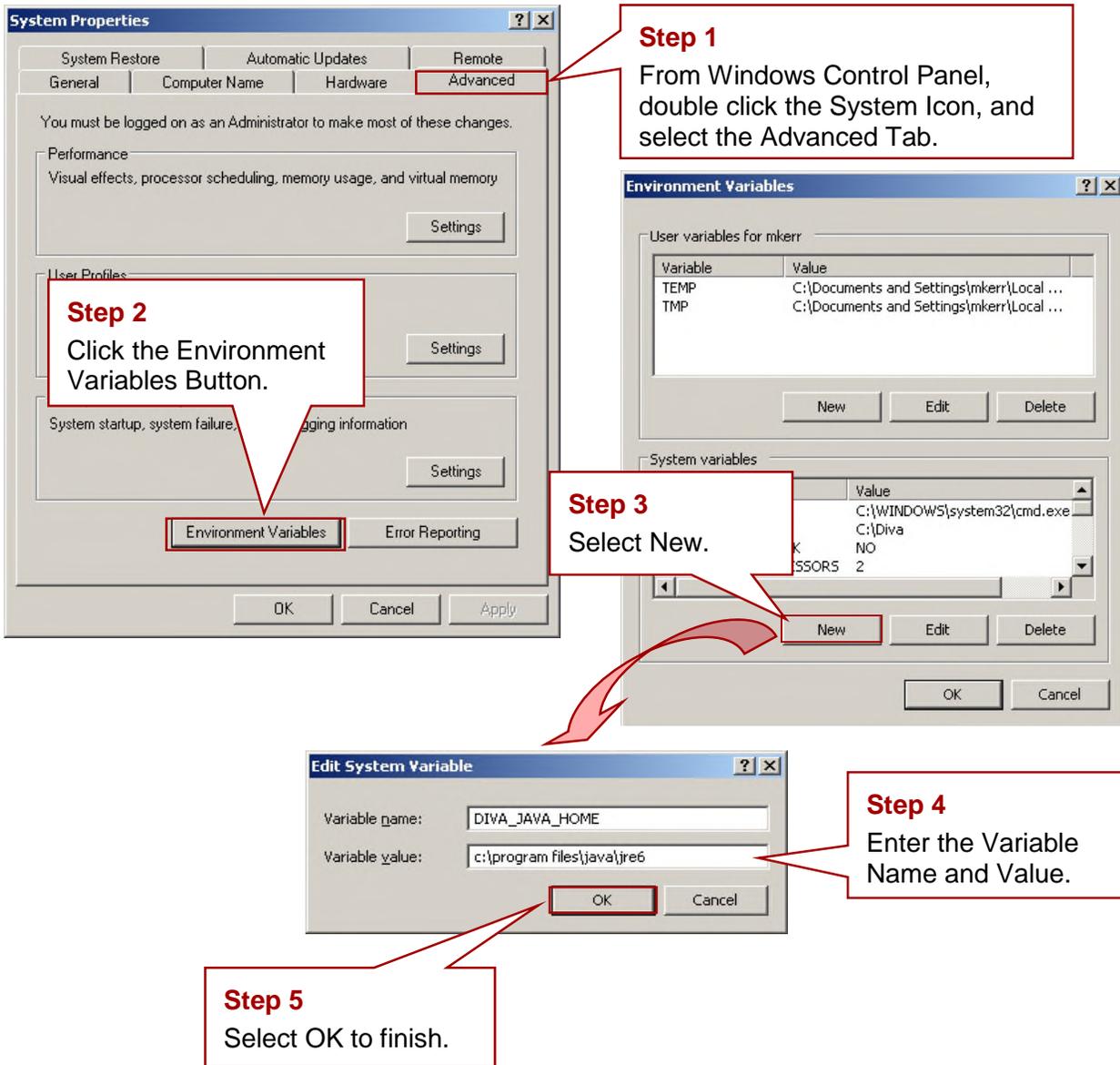
Some DIVArchive Software Components may require one or more Windows Operating System Environment Variables to be defined for those components to launch successfully.

Environmental variables allow the configured variable to be available to all programs rather than requiring it to be configured from the application itself each time it is run. A corollary of this feature is that it makes the variable independent of the application itself and therefore the variable's value does not need to be manually inserted or updated when the application software is updated or modified.

A **User Environmental Variable** indicates that the variable only applies to an individual Windows User Profile. A **System Environmental Variable** indicates it is applicable to all Windows User Profiles.

The following example illustrates how to configure the `DIVA_HOME` environment variable on a Windows system, which defines the path of the Java Runtime Environment for DIVArchive applications on the Windows host. This particular parameter for example, is required on any Windows host that will run either the DIVArchive Configuration Utility or the DIVArchive Control GUI.

Figure 14: Setting Windows Environment Variables



2.4 DIVArchive Configuration Utility Overview

The DIVArchive Configuration Utility connects primarily to the DIVArchive Database, and for some tasks, directly to the DIVArchive Robot Manager(s), if installed. After launching the utility you must first connect to the database to edit the DIVArchive System Configuration. The Oracle username and password for DIVArchive is arbitrary and can vary between installations, however it is typically `diva` and `lib5`. These types of details should be documented in the DIVArchive Delivery Plan for the customer's site.

The utility can be installed and run on any host with TCP/IP connectivity to the DIVArchive Database, Manager, and RobotManager(s). Since it is a Java based utility, a valid Java Runtime Environment (*JRE*) must also be installed on the host. For the Configuration Utility to launch, the `DIVA_JAVA_HOME` environmental variable in the host Operating System must also be defined. This variable should match the absolute directory path to the JRE `bin` folder; for example `DIVA_HOME\Java` (*where DIVA_HOME is the chosen DIVArchive installation directory*). Note that the required version of Java is delivered with DIVArchive, and installed during the DIVArchive installation process.

Note: The Configuration Utility is intended only for experienced users. Incorrect or incomplete changes in the Configuration Utility can adversely affect DIVArchive operations (*and possibly even delete data from the archive*), or prevent the DIVArchive Manager from running. If you are unsure about desired changes, you should contact Oracle Technical Support for assistance.

2.5 Connecting to the DIVArchive Database

Figure 15: Connecting to the DIVArchive Database

From the **File** pull-down menu, select **Connect**.

A quick alternative is to click the **Connect** icon.

The **DB Connection** window is displayed to establish a connection to the DIVArchive Database.

Oracle Username (typically *diva*).

Oracle System Identifier (Typically *lib5*).

Oracle Listener port (Typically *1521*).

Oracle password.

IP Address of the host where the DIVArchive Database is installed.

The connection status is indicated in the Configuration Utility notification area.

User: diva, connected to database on 192.168.2.100:1521.

If the connection fails, an error message will be generated in the notification area. In this case, the username or password was incorrect.

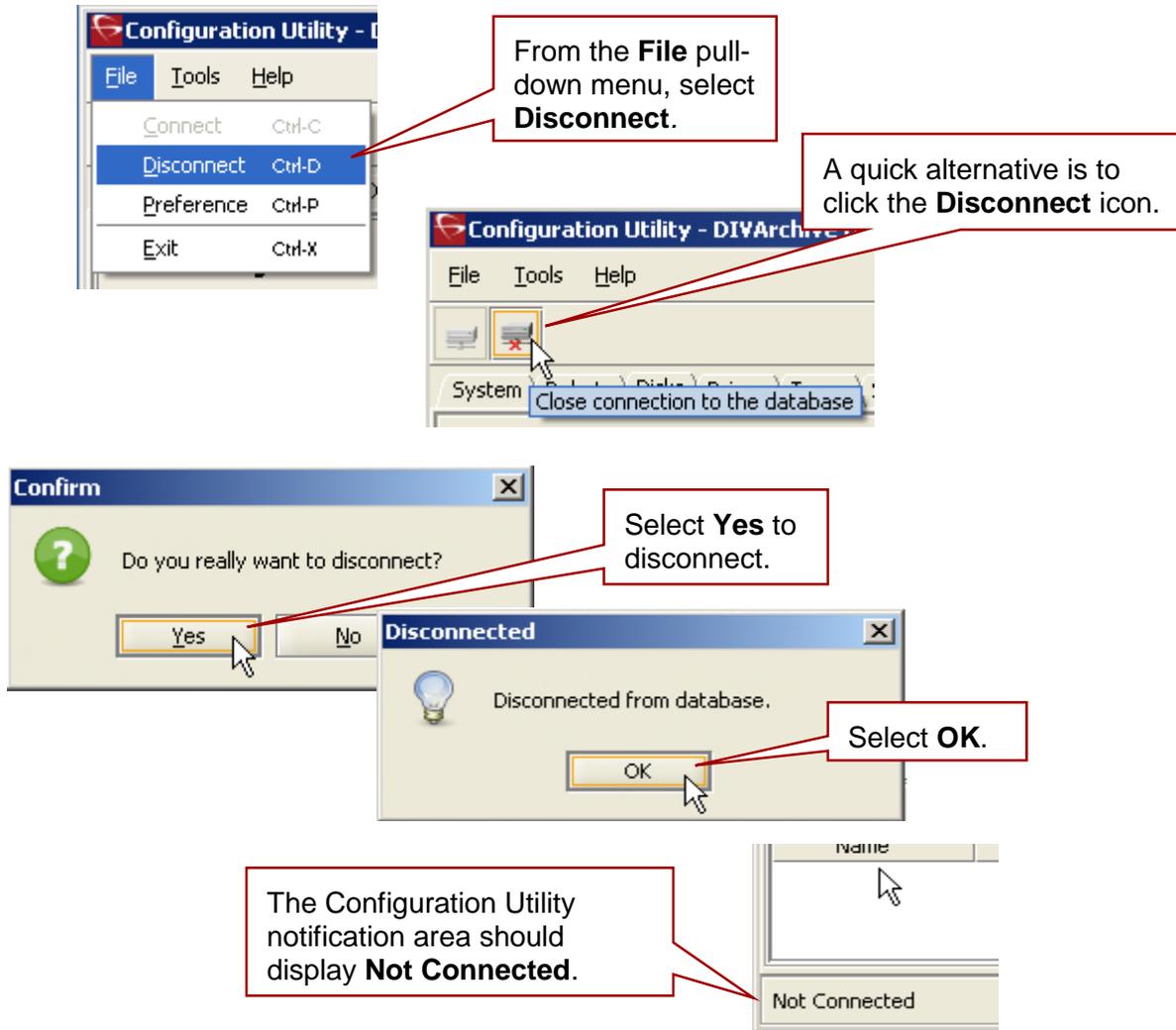
This is the error code returned from Oracle. If you still cannot connect, you should contact Oracle Technical Support for assistance.

Error: could not connect to DB : java.sql.SQLException: ORA-01017: invalid username/password; logon denied

2.6 Disconnecting from the DIVArchive Database

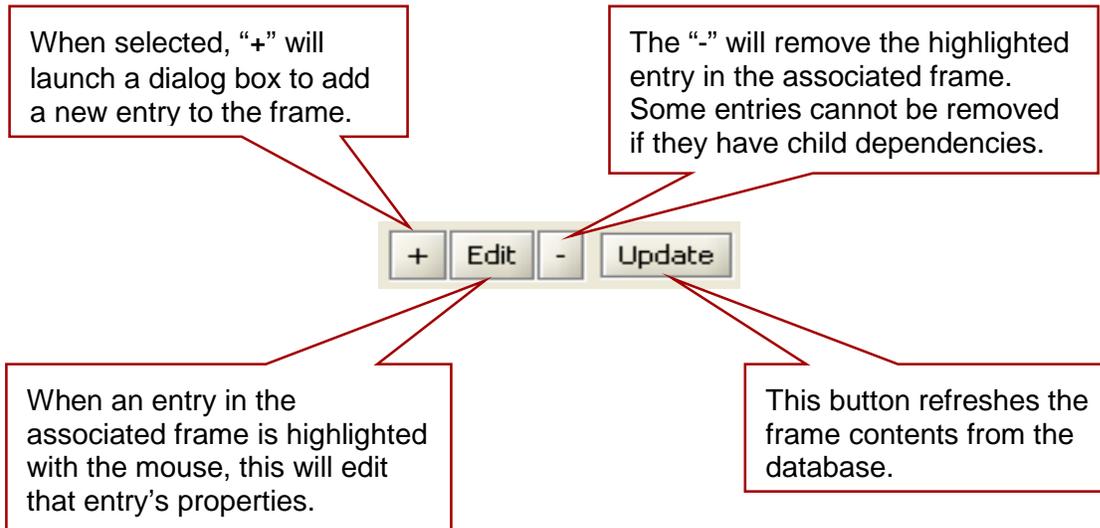
When the Configuration Utility is not in use, it should be disconnected from the DIVArchive Database.

Figure 16: Disconnecting from the DIVArchive Database



2.7 Frame Buttons

Figure 17: DIVArchive Frame Buttons



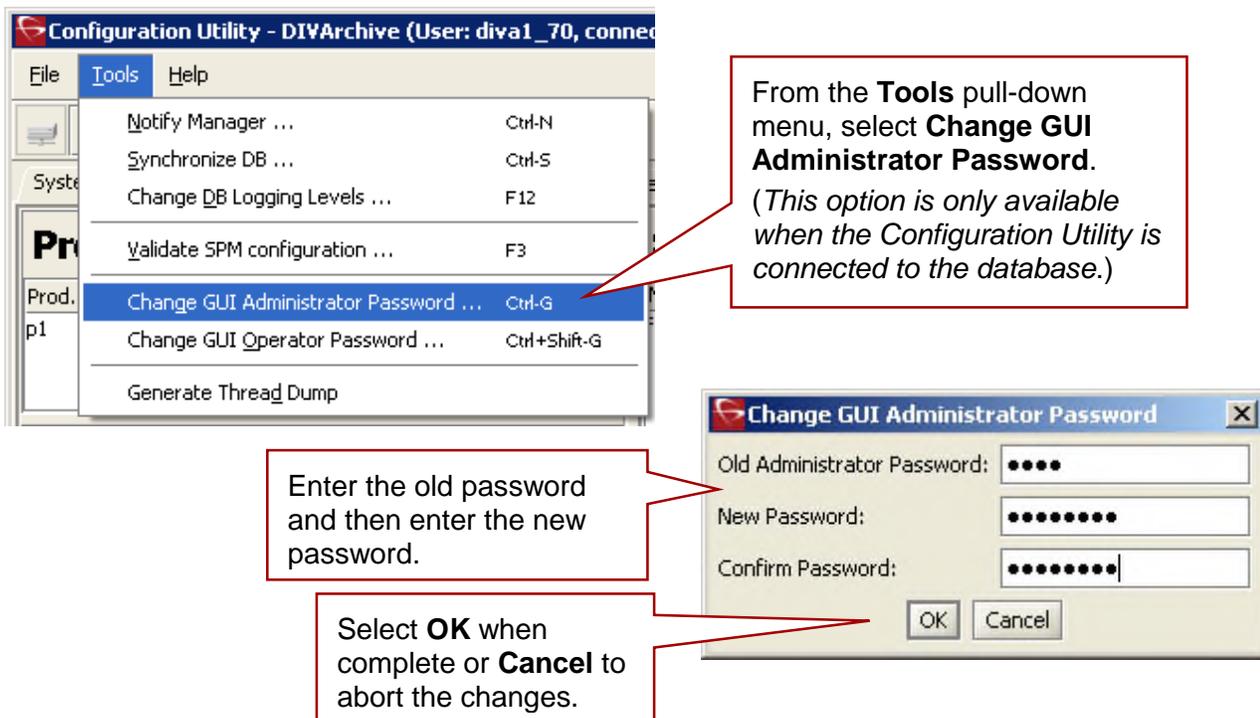
2.8 Control GUI Passwords

The DIVArchive Control GUI provides three fixed user profiles (*Administrator*, *Operator*, and *User*) to provide varying levels of access. The Administrator and Operator profiles both require a password that can be changed via the Configuration Utility. If the default passwords are not changed, it leaves the DIVArchive System accessible to possible malicious activity.

Note: The Administrator profile in the Control GUI allows access to all advanced functions within DIVArchive; restrictions may be implemented by incorporating the DIVArchive Access Gateway between the Control GUI and the DIVArchive Manager. It is **HIGHLY** recommended that the default passwords be changed immediately after installation and configuration for both the Administrator and Operator accounts! After the change has been made, please submit the new passwords to Oracle Support.

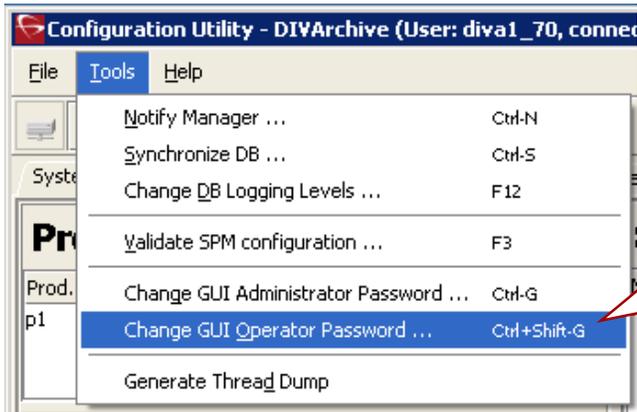
2.8.1 Setting the GUI Administrator Password

Figure 18: Setting the DIVArchive GUI Administrator Password



2.8.2 Setting the GUI Operator Password

Figure 19: Setting the GUI Operator Password



Configuration Utility - DIYArchive (User: diva1_70, connected)

File Tools Help

- Notify Manager ... Ctrl-N
- Synchronize DB ... Ctrl-S
- Change DB Logging Levels ... F12
- Validate SPM configuration ... F3
- Change GUI Administrator Password ... Ctrl-G
- Change GUI Operator Password ... Ctrl+Shift-G**
- Generate Thread Dump

From the **Tools** pull-down menu, select **Change GUI Operator Password**.
(This option is only available when the Configuration Utility is connected to the database.)



Change GUI Operator Password

Administrator Password: ●●●●

New Operator Password: ●●●●●●●●

Confirm Password: ●●●●●●●●

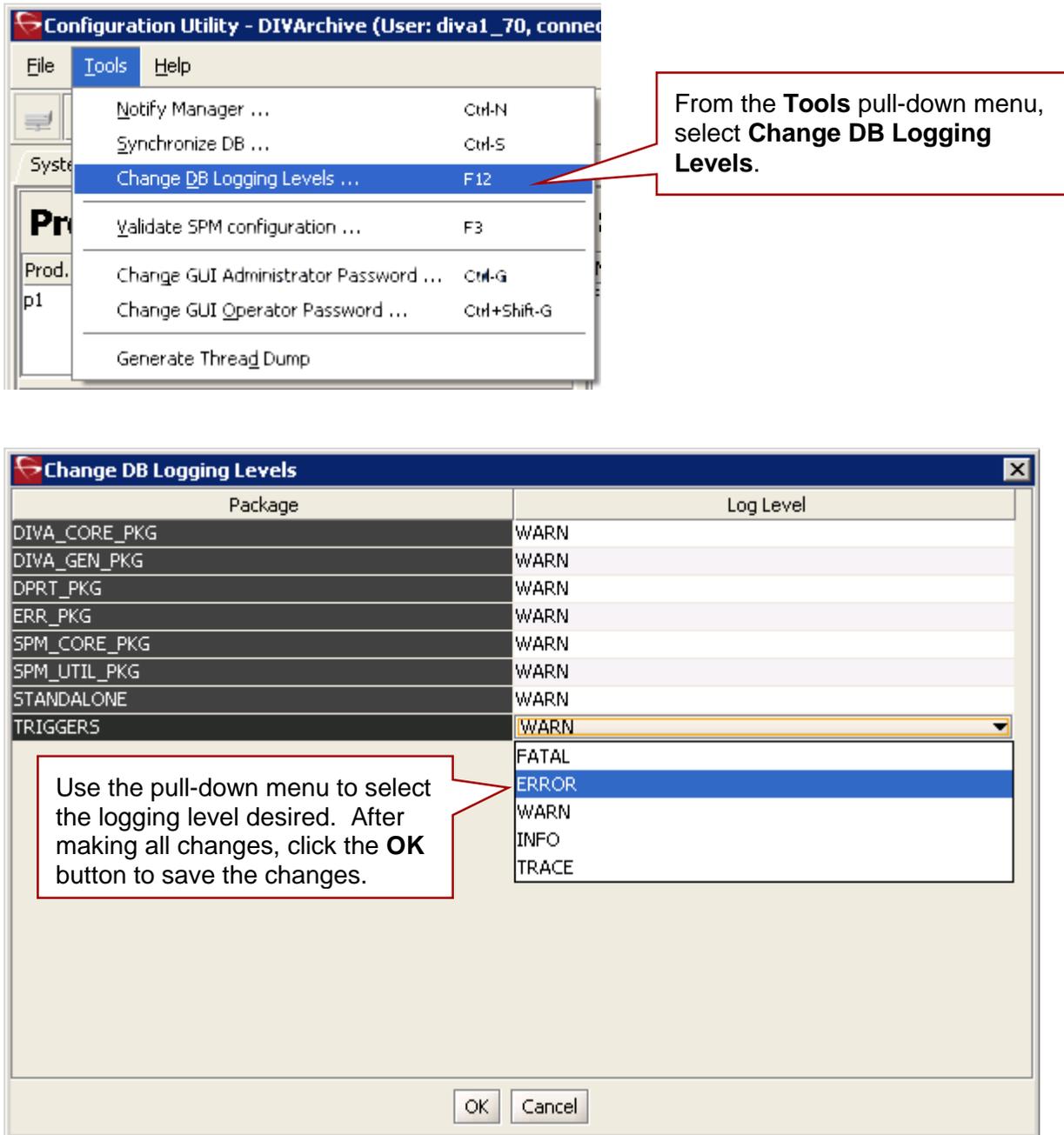
OK Cancel

Enter the old password and then enter the new password.

Select **OK** when complete or **Cancel** to abort the changes.

2.8.3 Changing the Database Log Level

Figure 20: Changing the DB Logging Levels



2.9 System Tab

The **System** tab defines key parameters for your DIVArchive installation and is the starting point for creating your DIVArchive configuration.

A drawing of the system components should be created including the data and control paths between them, how they will interact with each other, established naming conventions for resources such as disks, and the workflow of the platform before entering details into the Configuration Utility. Some parameters are difficult to change once they have dependencies from other configuration parameters in the database.

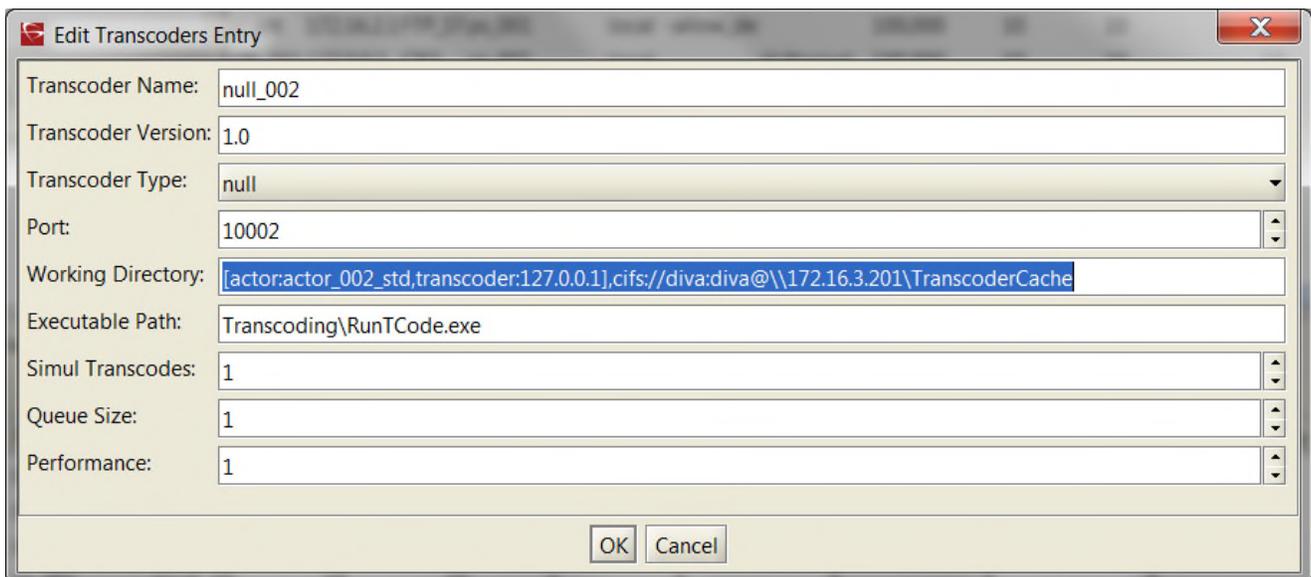
A transcoder is no longer coupled to a single actor. The transcoder is selected after the actor is selected; therefore users no longer need to define a LOCAL transcode actor as a destination. A local actor destination is dynamically and temporarily (*only in memory – not stored in the database*) created for the actor that is chosen as part of resource selection. The actor column was removed from the Transcoders area in the Configuration Utility. The transcoder server and cache location is now embedded in the **Working Directory** on the Transcoders Entry screen in the following format:

```
[actor:actor_name,transcoder:transcoder_ip_address],cifs://username:password@\\transcoder_cache_ip_address\transcoder_cache"
```

Notes:

- **Multiple transcoders are not supported for Flip Factory. They are only supported for Vantage.**
- **The original method of configuring a transcoder to a local Actor is still supported for legacy purposes.**
- **The original method of configuring Local Sources/Destinations tied to actors is still supported so legacy configurations will continue to function.**
- **If a transcoder address is not specified in the transcoder's working directory, a local transcoder address of 127.0.0.1 will be assumed as the transcoder address.**

Figure 21: Remote Working Directory Sample Entry

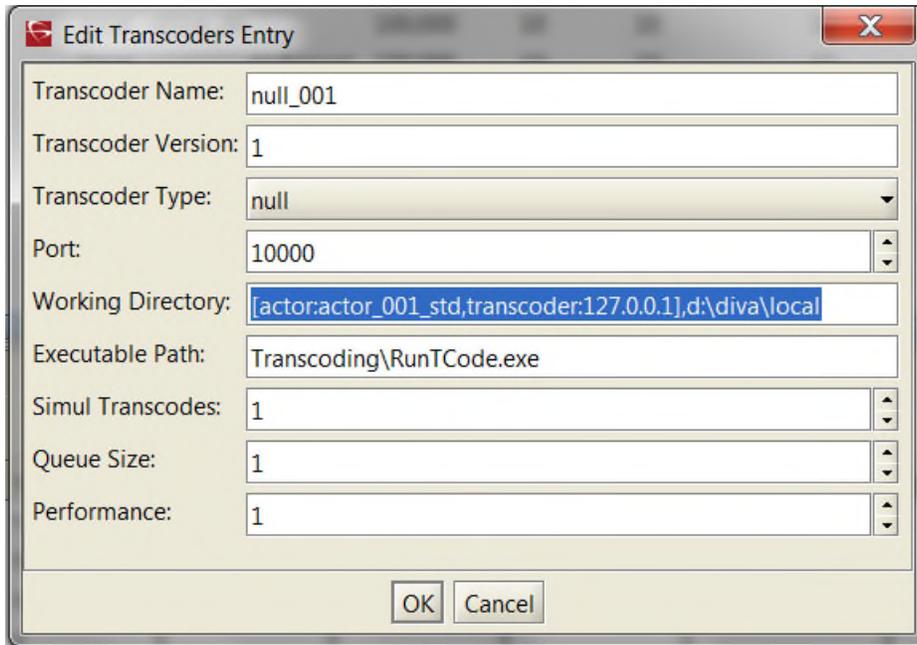


The screenshot shows a dialog box titled "Edit Transcoders Entry" with the following fields and values:

Transcoder Name:	null_002
Transcoder Version:	1.0
Transcoder Type:	null
Port:	10002
Working Directory:	[actor:actor_002_std,transcoder:127.0.0.1],cifs://diva:diva@\\172.16.3.201\TranscoderCache
Executable Path:	Transcoding\RunTCode.exe
Simul Transcodes:	1
Queue Size:	1
Performance:	1

At the bottom of the dialog are "OK" and "Cancel" buttons.

Figure 22: Local Working Directory Sample Entry



To notify the actors of any changes in the actor configuration, click on **Notification**, **Notify Actors** while connected to the Manager. The actors must be running and connected to the Manager to receive the notifications.

Figure 23: Notifying Actor of Configuration Changes

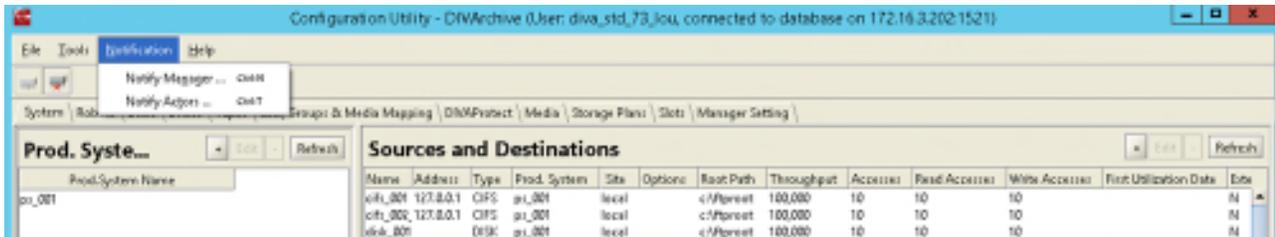


Figure 24: DIVArchive System Tab Display Window

Production System Definitions: All DIVArchive installations have at least one production system. Additional production systems allow dedication of a particular Actor for specific destinations.

Sources and Destinations Definitions: These define where DIVArchive will archive from (*Source*) and restore to (*Destination*).

Site Definitions: All installations have at least one site. Additional sites are optional and may be considered by the DIVArchive Manager for optimal resource allocation.

Note: Site Support must be enabled in the DIVArchive Manager configuration, otherwise all sites will be considered equally.

DIVArchive Actor Host Definitions and Logical Functions: All installations have at least one Actor.

DIVArchive Transcoder and Analyzer Definitions: Automatically chooses the Actor(s) either attached to a Bitstream/Flip Factory Transcoder installation or integrated with the DIVAnalyze Harris QuiC compressed file analysis software. DIVArchive allows a single transcoder to perform multiple transcodings. DIVArchive assigns additional ports as needed from the base port specified in the configuration. Therefore, a gap of 100 between individual transcoder port settings is recommended to avoid port conflicts.

The transcoder server and cache location is now embedded in the **Working Directory** on the Transcoders Entry screen in the following format:

```
[actor:actor_name,transcoder:transcoder_ip_address],cifs://username:password@\\transcoder_cache_ip_address\transcoder_cache
```

If a transcoder address is not specified in the transcoder's working directory, a local transcoder address of 127.0.0.1 will be assumed as the transcoder address.

Note: Multiple transcoders are not supported for Flip Factory. They are only supported for Vantage.

2.9.1 Actor Configuration in the Database

With the exception of the **Service Name** and **Port**, all actor configuration settings have been moved from the Actor Configuration and Partial Restore Configuration files to the Configuration Utility. These settings are now located under **Actor Advanced** and **Partial Restore Settings Tabs** of the **Actor Panel** of the **Systems Tab**. Some settings are only available In Engineering Mode and are labeled with an X in the **Engineering Mode** column of the tables below.

Table 4: Advanced Actor Parameters

Name	Type	Minimum	Maximum	Engineering Mode	Default
DISABLE DISK PREALLOCATION	BOOLEAN			X	YES
TAPE TEST UNIT READY TIMEOUT (S)	INTEGER	60	1200		180
DO NOT CHECK OBJECT NAME	BOOLEAN			X	NO
DO NOT CHECK CATEGORY	BOOLEAN			X	NO
SIMULATION	STRING			X	
SIMULATION READING ERROR RATE (%)	INTEGER	0	100	X	0
SIMULATION WRITING ERROR RATE (%)	INTEGER	0	100	X	0
SIMULATION TAPE SIZE (MB)	INTEGER	20	500000	X	300000
SEACHANGE CHECK DELAY (MS)	INTEGER	0	10000	X	1000
PROFILE READ BLOCK SIZE (B)	INTEGER	1500	262144		1500
PROFILE WRITE BLOCK SIZE (B)	INTEGER	1500	262144		32768
QUANTEL RENAME CLIPS	BOOLEAN				NO
QT SELF-CONTAINED THRESHOLD (MB)	INTEGER	10	100		50
RENAME_TRANSCODED_CLIPS	BOOLEAN			X	
DIRECTORY SERVER ENABLED	BOOLEAN			X	YES
DISK FTP PASSIVE MODE	BOOLEAN				NO
DISK FTP BLOCK SIZE	INTEGER	1024	524288		32

Name	Type	Minimum	Maximum	Engineering Mode	Default
DISK FTP SOCKET WINDOW SIZE (B)	INTEGER	65536	10485760		65536

Table 5: Partial File Restore Parameters

Name	Type	Minimum	Maximum	Engineering Mode	Default
QT IGNORE START TIMECODE	BOOLEAN				NO
QT OMNEON FIRST FRAME HANDLING	STRING				RESET
AVI IGNORE START TIMECODE	BOOLEAN				NO
EVS MXF IGNORE START TIMECODE	BOOLEAN				NO
GXF TIMECODE REFERENCE	INTEGER	0	2		1
GXF PROGRESSIVE TIMECODE TRANSLATION	BOOLEAN				NO
LXF IGNORE START TIMECODE	STRING				NO
MXF PARTIAL RESTORE DICTIONARY_FILE	STRING				
MXF TIMECODE FROM SOURCE PACKAGE	BOOLEAN				NO
MXF TIMECODE VALUE TO SWITCH PACKAGE	STRING				-1
MXF ENFORCE CLOSED HEADER	STRING				YES
MXF RUN IN PROCESSOR	STRING				
MXF IGNORE START TIMECODE	BOOLEAN				NO
MXF USE OMNEON DARK METADATA	BOOLEAN				NO
MXF SERIALIZE DEPTH FIRST	BOOLEAN				NO

Name	Type	Minimum	Maximum	Engineering Mode	Default
MXF GENERATE RANDOM INDEX PACK	BOOLEAN				YES
MXF NUMBER FRAMES PER BODY PARTITION	INTEGER	50	500		250
MXF UPDATE TCTRACK ORIGIN	BOOLEAN				NO
MXF TOLERANCE ON TCOUT	INTEGER	0	250		0
MXF DURATION FROM FOOTER	BOOLEAN				YES
MXF MAX QUEUE SIZE	INTEGER	100	1000		200
SEACHANGE IGNORE START TIMECODE	BOOLEAN				NO
MPEG2 TRANSPORT STREAM IGNORE START TIMECODE	BOOLEAN				NO
MPEG2 PROGRAM STREAM IGNORE START TIMECODE	BOOLEAN				NO

2.10 Robots Tab

The **Robots** tab is present in all DIVArchive installations (*although not every installation necessarily has a library*). It defines basic associations with the Robotics software and hardware components.

Figure 25: DIVArchive Robots Tab Window

The screenshot shows the 'Robots' tab in the DIVArchive Configuration Utility. The window title is 'Configuration Utility - DIVArchive (User: diva1_70, connected to database on 127.0.0.1:1521)'. The 'Robots' tab is selected in the menu bar. The window is divided into four main sections: 'Robot Managers', 'Libraries', 'Media Compatibility', and 'Robot Managers-ACS'. Each section contains a table of data and has a callout box explaining its function.

Robot Managers

Robot Manager Name	Address	Port	Site
robot1_70	127.0.0.1	8501	local

This frame defines to DIVArchive the connection parameters to each host running a DIVArchive Robot Manager Instance.

Libraries

ACS	NAME	Serial Number	Type	Status	First Utilization Date
0	ACS_0	123456789	First ACS	Online	2012-04-11 20:13:50.0

Displays the tape or DVD libraries currently configured via one or more DIVArchive Robot Managers and their online status.

Media Compatibility

Media Type ID	Drive Type ID	Compatibility Type
LTO-100G	simu_IBM_LTO	READ-WRITE
LTO2-100G	simu_IBM_LTO2	READ-WRITE

This frame maps the Tape Media Type defined in the **Tapes** tab, to the Drive Types defined in the **Drives** tab. Although entries in this frame can be manually removed, they can only be added or updated during a Database Synchronization with a Robot Manager.

Robot Managers-ACS

Robot Manager Name	ACS
robot1_70	0

This frame associates each Robot Manager with an Automated Cartridge System (ACS) number. Although entries in this frame can be manually removed, they can only be added by performing a Database Synchronization with the specific Robot Manager.

2.11 Disks Tab

The **Disks** tab defines the physical disks that are to be used by DIVArchive, how they are grouped together for either permanent or cache storage, and how each disk is logically accessed by the Actors.

Figure 26: DIVArchive Disks Tab Window

The screenshot shows the 'Disks' tab in the Configuration Utility. It is divided into three main sections: Arrays, Disks, and Actor-Disk Connections. Callouts provide explanations for each section.

Arrays: An Array defines a logical association of disks in which one or more physical disks are assigned for use by DIVArchive. The Array Name is equivalent to the Group Name for a tape.

Name	Description	Format	Max Allowed Disk Space For Repack (G)
array_axf		AXF	75
array_axf_2		AXF	75
array_axf_vw		AXF	50
array_leg		Legacy	75
array_leg_vw		Legacy	50

Disks: The symbolic name and location for each disk in your system, whether confined to a single host or shared between hosts. These disks are then assigned to Arrays.

Name	Array	Site	Status	Min.space	VW
disk_axf	array_axf	local	Online	0	OFF
disk_axf_2	array_axf_2	local	Online	0	OFF
disk_axf_vw	array_axf_vw	local	Online	0	ON
disk_leg	array_leg	local	Online	0	OFF
disk_leg_vw	array_leg_vw	local	Online	0	ON

Actor-Disk Connections: Configures how each disk is logically connected to each DIVArchive Actor, and how it is to be used. For shared disks accessible by more than one Actor, the disk connection must be declared for all Actors.

Disk Id	Disk	Actor	Interface	Mount point	Throughput	Access	Usage
1	disk_axf	actor1_70	local	D:\diva\70\simulator\disks\disk1	100,000	Read/Write	CACHE_AND_STORAGE_AND_NEARLINE
3	disk_axf_vw	actor1_70	local	D:\diva\70\simulator\disks\disk1	100,000	Read/Write	CACHE_AND_STORAGE
4	disk_leg	actor1_70	local	D:\diva\70\simulator\disks\disk2	100,000	Read/Write	CACHE_AND_STORAGE_AND_NEARLINE
6	disk_leg_vw	actor1_70	local	D:\diva\70\simulator\disks\disk2	100,000	Read/Write	CACHE_AND_STORAGE
15	disk_axf_2	actor2_70	remote	C:\diva\70\simulator\disks\disk1	100,000	Read/Write	CACHE_AND_STORAGE_AND_NEARLINE

User: diva1_70, connected to database on 127.0.0.1:1521.

2.12 Drives Tab

The **Drives** tab is where the drives in your tape library(s) are identified and configured for use with DIVArchive and its Actors. In some installations, a tape library and its drives may be shared with other applications and the configuration options allow you to disable any of the identified drives from DIVArchive use.

Figure 27: DIVArchive Drives Tab Window

Displays the drives currently identified to DIVArchive in a database synchronization and their current status.

ID	Serial Number	Firmware Version	ACS	LSM	Drive Type	Status	Operations	Used	Internal Name	Installation Date	Last Upgrade Date	Last Cle
0	SIMU000000	79M0	0	0	109	Online	A	Y	0-0-0	11/04/2012 20:02:38	11/04/2012 20:13:51	
1	SIMU000001	79M0	0	0	109	Online	A	Y	0-0-1	11/04/2012 20:02:38	11/04/2012 20:13:51	
2	SIMU000002	79M0	0	0	109	Online	A	Y	0-0-2	11/04/2012 20:02:38	11/04/2012 20:13:51	
3	SIMU000003	79M0	0	0	109	Online	A	Y	0-0-3	11/04/2012 20:02:38	11/04/2012 20:13:51	
4	SIMU000004	79M0	0	0	109	Online	A	Y	0-0-4	11/04/2012 20:02:38	11/04/2012 20:13:51	
5	SIMU000005	79M0	0	0	109	Online	A	Y	0-0-5	11/04/2012 20:02:38	11/04/2012 20:13:51	
6	SIMU000006	79M0	0	0	110	Online	A	Y	0-0-6	11/04/2012 20:02:38	11/04/2012 20:13:51	
7	SIMU000007	79M0	0	0	110	Online	A	Y	0-0-7	11/04/2012 20:02:38	11/04/2012 20:13:51	
8	SIMU000008	79M0	0	0	110	Online	A	Y	0-0-8	11/04/2012 20:02:38	11/04/2012 20:13:51	

Drive Properties

Drive Type	Drive Type Id
simu_ibm_lto	109
simu_ibm_lto2	110

Actors-Drives

Actor	Drive
actor1_70	0 (ACS=0, LSM=0)
actor1_70	1 (ACS=0, LSM=0)
actor1_70	2 (ACS=0, LSM=0)
actor1_70	3 (ACS=0, LSM=0)
actor1_70	4 (ACS=0, LSM=0)
actor1_70	5 (ACS=0, LSM=0)
actor1_70	6 (ACS=0, LSM=0)
actor1_70	7 (ACS=0, LSM=0)
actor1_70	8 (ACS=0, LSM=0)
actor1_70	9 (ACS=0, LSM=0)

This displays the drive models currently configured for use with DIVArchive. Although entries in this frame can be manually removed, they can only be added by performing a Database Synchronization with a Robot Manager.

Indicates to DIVArchive which Actors have access to the drives configured in the **Drives** frame.

2.12.1 Drive Edit Dialog Window

The **Drive Edit Dialog** allows the user to edit the serial number of a drive. This is useful if this information wasn't retrieved, or entered improperly, during a Sync DB process. The firmware of the drive is also displayed in a non-editable field (*this information is obtained from the Actors when they scan for tape drive devices*).

Figure 28: Drive Edit Dialog

Edit Drives Entry

Drive ID: 0

Serial Number: SIMU000000

Firmware Version: 79M0

ACS ID: 0

LSM ID: 0

Drive Type ID: 109

Status: Online

Enabled Operations: A

Used: Y

Internal Name: 0-0-0

Installation Date: 11/04/2012 20:02:38

Last Upgrade Date: 11/04/2012 20:13:51

Last Cleaning Date:

OK Cancel

This information is visible in the *Drives* panel as shown below:

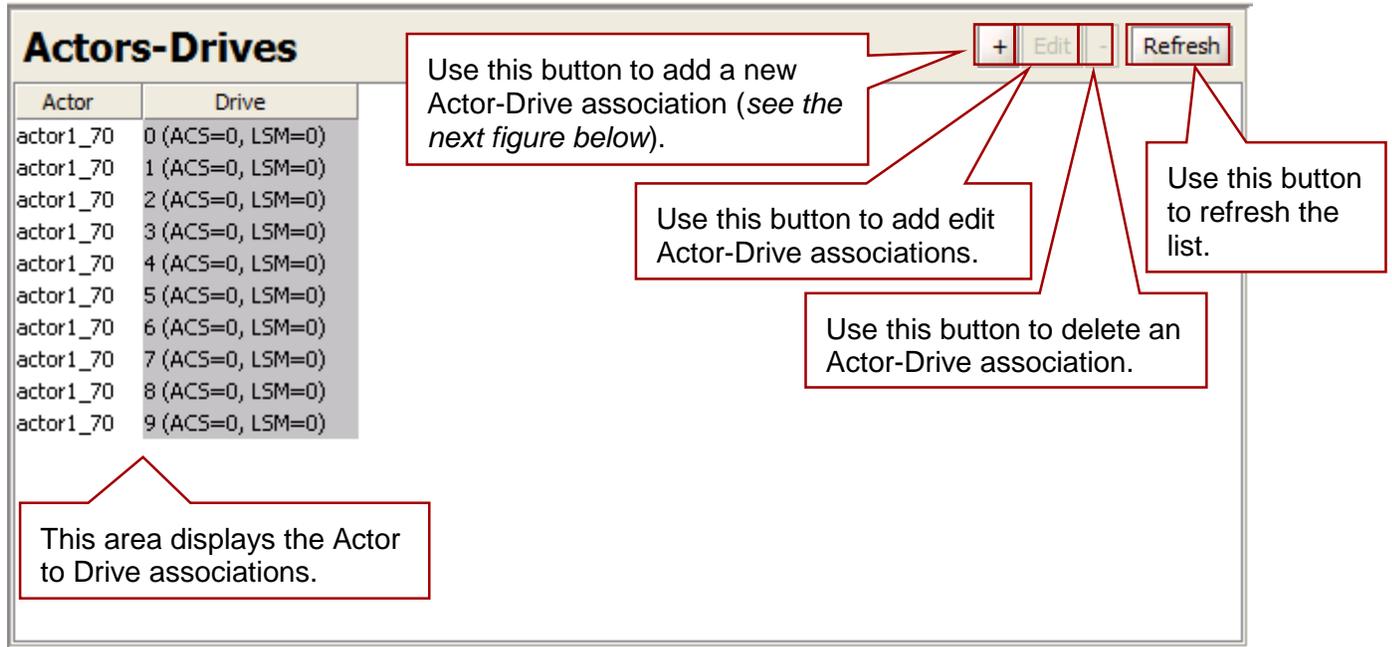
Figure 29: Drives Panel

ID	Serial Number	Firmware Version	ACS	LSM	Drive Type	Status	Operations	Used	Internal Name	Installation Date	Last Upgrade Date	Last Cle
0	SIMU000000	79M0	0	0	109	Online	A	Y	0-0-0	11/04/2012 20:02:38	11/04/2012 20:13:51	
1	SIMU000001	79M0	0	0	109	Online	A	Y	0-0-1	11/04/2012 20:02:38	11/04/2012 20:13:51	
2	SIMU000002	79M0	0	0	109	Online	A	Y	0-0-2	11/04/2012 20:02:38	11/04/2012 20:13:51	
3	SIMU000003	79M0	0	0	109	Online	A	Y	0-0-3	11/04/2012 20:02:38	11/04/2012 20:13:51	
4	SIMU000004	79M0	0	0	109	Online	A	Y	0-0-4	11/04/2012 20:02:38	11/04/2012 20:13:51	
5	SIMU000005	79M0	0	0	109	Online	A	Y	0-0-5	11/04/2012 20:02:38	11/04/2012 20:13:51	
6	SIMU000006	79M0	0	0	110	Online	A	Y	0-0-6	11/04/2012 20:02:38	11/04/2012 20:13:51	
7	SIMU000007	79M0	0	0	110	Online	A	Y	0-0-7	11/04/2012 20:02:38	11/04/2012 20:13:51	
8	SIMU000008	79M0	0	0	110	Online	A	Y	0-0-8	11/04/2012 20:02:38	11/04/2012 20:13:51	

2.12.2 Actors-Drives Area

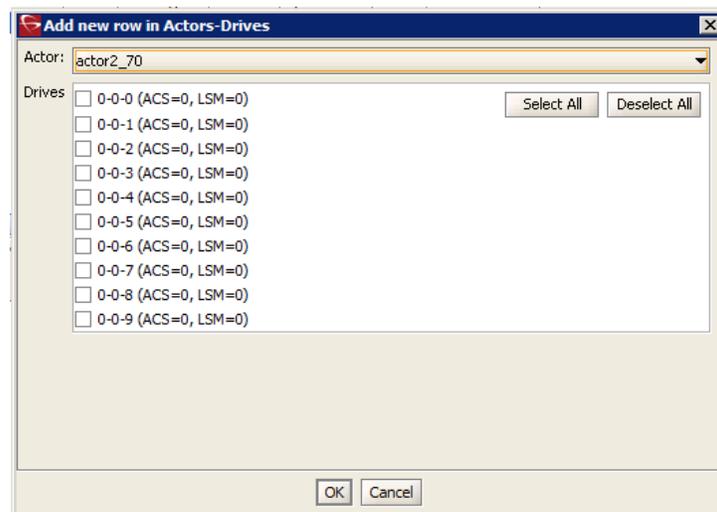
The Actors-Drives Area displays the Actor to Drive associations. In this area associations may be added, edited, or deleted.

Figure 30: Actors-Drives Area Close-up



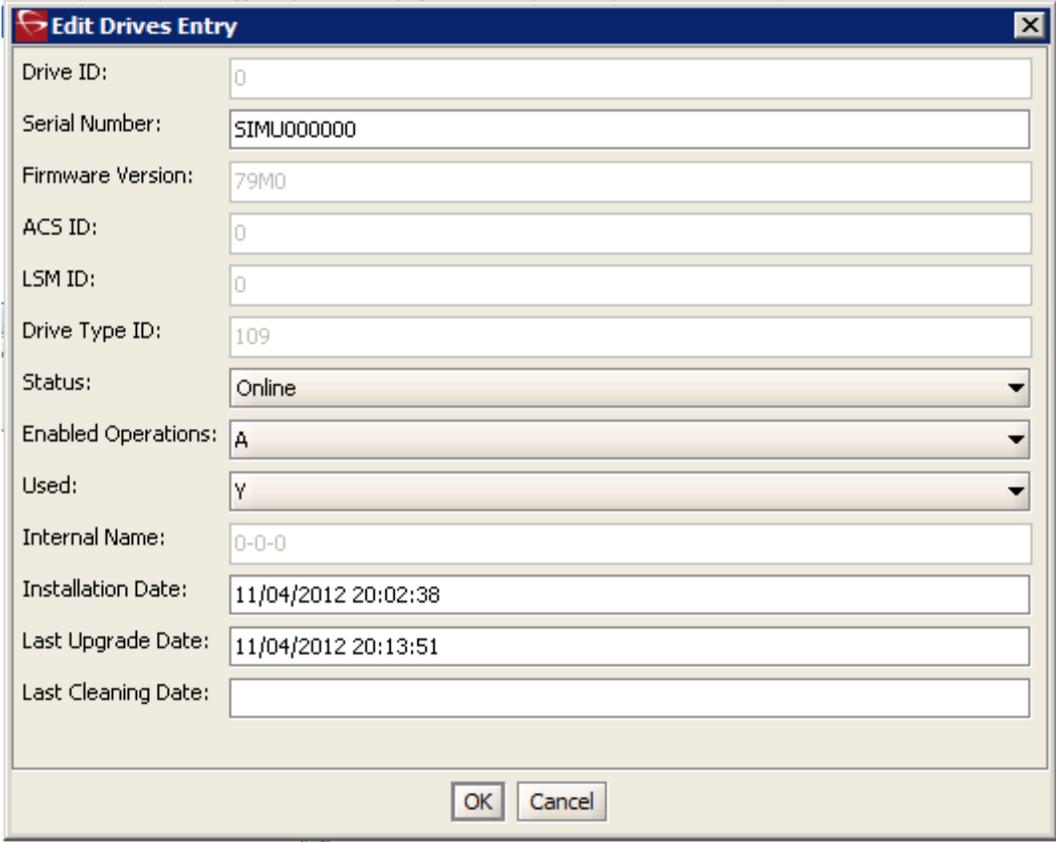
Click on the “+” (*plus*) button to add a new existing Actor-Drive Association. The Add New Row in Actors-Drives window will be displayed. Use the pull-down box to select the Actor and then use the checkboxes next to each drive to be associated with that Actor.

Figure 31: Edit Actors-Drives Window



Click on the **Edit** Button to edit an existing Actors-Drives association. The **Edit Drives Entry** window is displayed. Make the required/desired updates and click the **OK** Button to save the changes.

Figure 32: Edit Drives Entry Window



The screenshot shows a window titled "Edit Drives Entry" with a close button (X) in the top right corner. The window contains the following fields and values:

Drive ID:	0
Serial Number:	SIMU000000
Firmware Version:	79M0
ACS ID:	0
LSM ID:	0
Drive Type ID:	109
Status:	Online
Enabled Operations:	A
Used:	Y
Internal Name:	0-0-0
Installation Date:	11/04/2012 20:02:38
Last Upgrade Date:	11/04/2012 20:13:51
Last Cleaning Date:	

At the bottom of the window, there are two buttons: "OK" and "Cancel".

2.13 Tapes Tab

The **Tapes** tab is for the definition of each Tape Media Type capacity in DIVArchive, along with each individual tape's write, repack or "to be cleared" status. Tapes that do not contain any DIVArchive Objects (*i.e. are empty or are from another archive application in a shared library environment*) and have been ejected from a DIVArchive managed library can also be deleted from the DIVArchive database in this tab.

Figure 33: DIVArchive Tapes Tab Window

The screenshot shows the DIVArchive Tapes Tab window with the following sections and callouts:

- Tape Properties:** Displays the Tape Types and configuration parameters currently configured in DIVArchive after a library Database Synchronization. The table below shows the data:

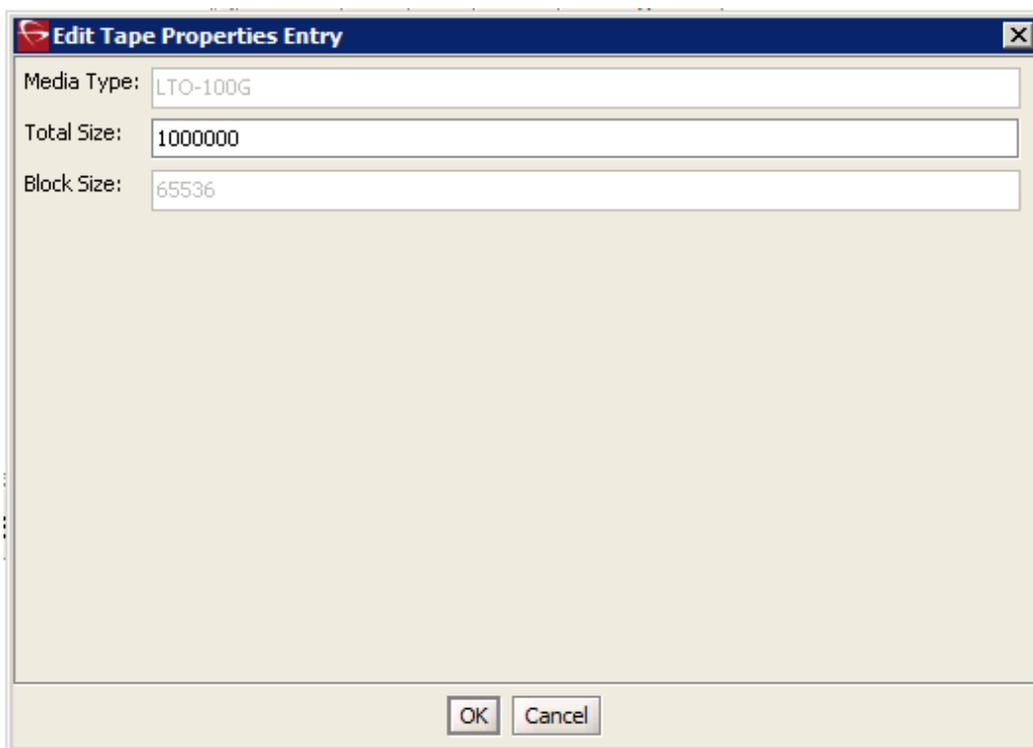
Media Type	Total Size	Block Size
LTO-100G	1,000,000	65,536
LTO2-100G	1,000,000	65,536

- Empty Ejected Tapes:** Displays the tapes that no longer have any DIVArchive content and have been ejected from an attached library. A callout points to a red 'X' icon in the top right corner, stating: "This button will permanently remove the selected tape from the DIVArchive Database."
- Inserted Protected Tapes:** When a tape is externalized, it is set to **Protected Mode** by DIVArchive. This state needs to be manually removed after reinsertion into the library if the tape is to have new material written to it.
- Tape States:** A tape will appear in this frame if either the **Enable for Writing** or the **Enable for Repack** states is set to "N". The **Enable for Writing** state may be automatically disabled by DIVArchive if it encounters an error during a read, write or repack operation.

Note: The Tape States Frame gives an overall indication of the reliability of your tape drives. Tapes appearing in this frame (*if not manually inserted*) indicates that either a read or write error occurred on that tape during DIVArchive operations. If you have a large number of tapes present here this may indicate an issue with one or more of your tape drives and should be promptly investigated.

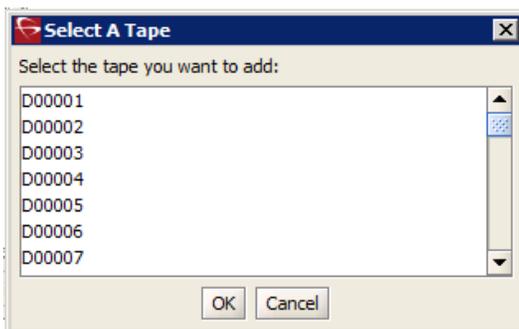
In the **Tape Properties Area**, highlight an existing tape and click the **Edit** Button to open the Tape Properties window.

Figure 34: Edit Tape Properties Window



Click on the “+” (*plus*) Button in the **Tape States Area** to add a new tape to the Tape States area. Select the tape to add and click the **OK** Button.

Figure 35: Add New Tape State Window



2.14 Sets, Groups and Media Mapping Tab

The **Sets, Groups & Media Mapping** tab is used to allocate new tapes into pools for use by DIVArchive. Each media pool is represented by its **Set ID**. The Set ID is typically used to distinguish different types of tape media; however, it may also be used to dedicate a specific set of tapes to specific groups.

A **Group** is a logical name for the storage of DIVArchive Objects. Each group is assigned a Set ID of tapes to draw upon. Each group can only be assigned one Set ID. Several groups can share the same Set ID.

Figure 36: DIVArchive Sets, Groups & Media Tab Window

The screenshot shows the 'Sets, Groups & Media Mapping' window. It is divided into three main sections: 'Unused Tapes Sets', 'Groups', and 'Media Mapping'. The 'Unused Tapes Sets' section displays a table of tapes with columns for Barcode, ACS, LSM, Media Type, and Set ID. The 'Groups' section displays a table of groups with columns for Id, Group Name, Set ID, Description, Media Types, Tape Format, Worse Fit Enable, Repack Reservation, and VW. The 'Media Mapping' section displays a table with columns for Id, Name, From, Map to Media, and Map to Storage Plan. Callouts provide detailed explanations for each section.

Displays empty tapes that are recognized by DIVArchive and the library module where they are located. The Set ID of each tape can also be defined in this frame.

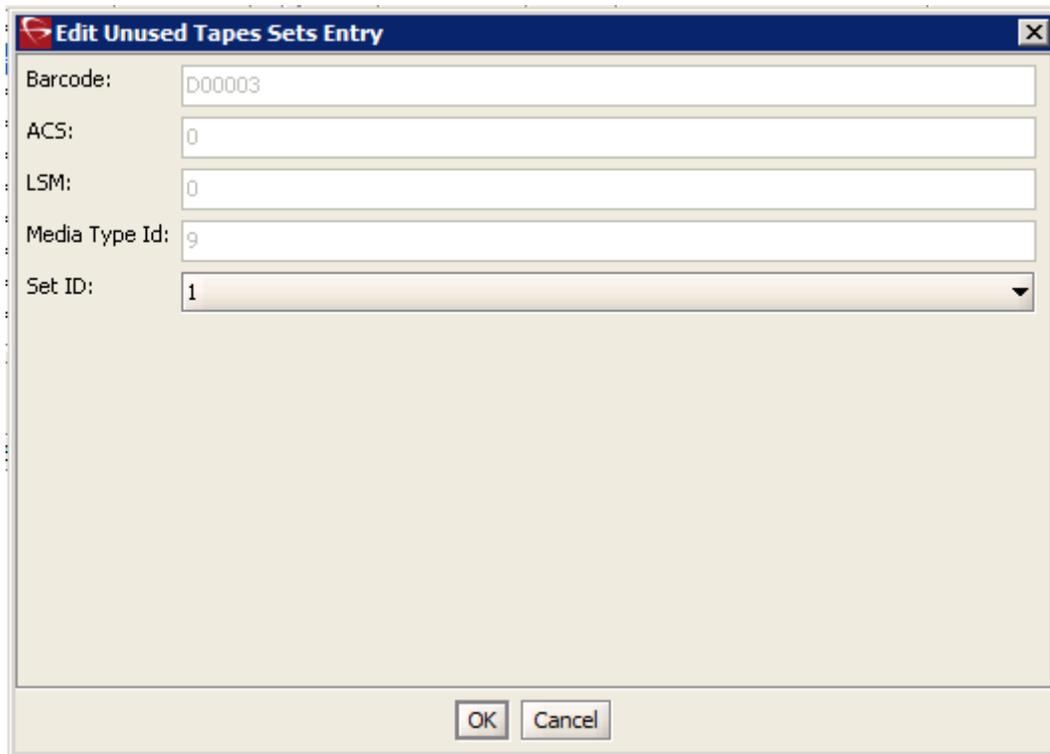
Adds, removes or edits existing groups, and each group's association with the tape pools defined in the **Unused Tapes Sets Frame**.
Note: A group can only be removed when it no longer contains any DIVArchive Objects.

Additional Set IDs for the **Unused Tape Sets Frame** are only available once they are first created in a group. Tapes that should not be used by DIVArchive should be configured with a Set ID of 99.

Media Mapping allows DIVArchive to automatically alter the specified media in an archive request to another Disk Array, Tape Group or Storage Plan. In this way, the storage for archive requests can be altered without requiring any changes in the archive initiator (*automation or MAM system*).

In the Unused Tape Sets Area, highlight an existing tape and click on the Edit Button to display the Edit Unused Tapes Sets Entry Window. When done editing the Tape Set click the Refresh Button to refresh the list.

Figure 37: Edit Unused Tapes Sets Entry Window



The screenshot shows a dialog box titled "Edit Unused Tapes Sets Entry". It contains the following fields and values:

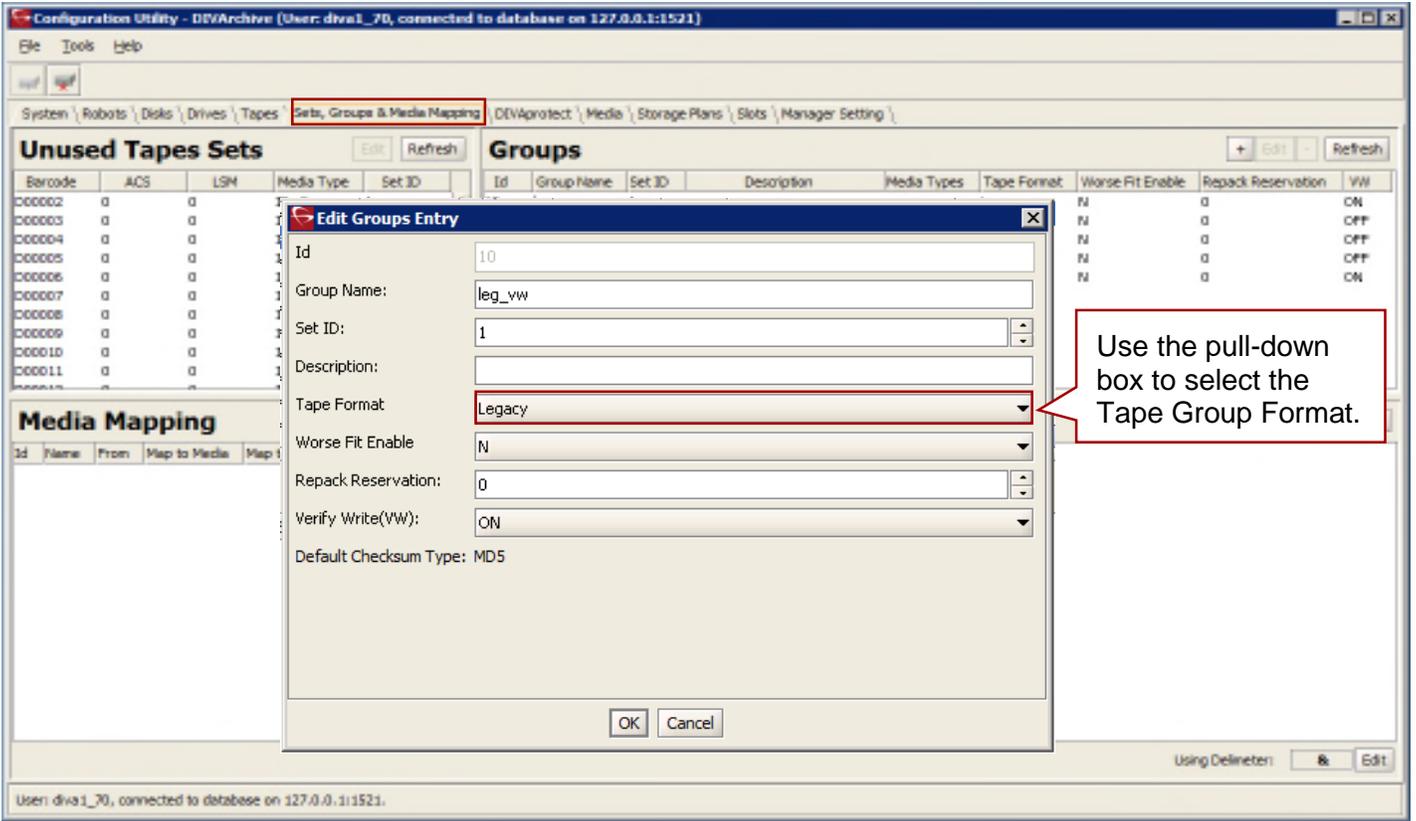
Field	Value
Barcode:	D00003
ACS:	0
LSM:	0
Media Type Id:	9
Set ID:	1

At the bottom of the dialog are "OK" and "Cancel" buttons.

Using the DIVArchive Version 7.3 Configuration Utility, the user can define the format of an array or group. The format is configured via the **Disks** and **Sets, Groups & Media Mapping** panels for Array and Groups respectively. Alternatively, the `addGroup` API call can be used to define a group or array and its format. The default format is AXF. This can also be achieved by selecting *Legacy* via the Configuration Utility, or specify the corresponding value for the format via the API call.

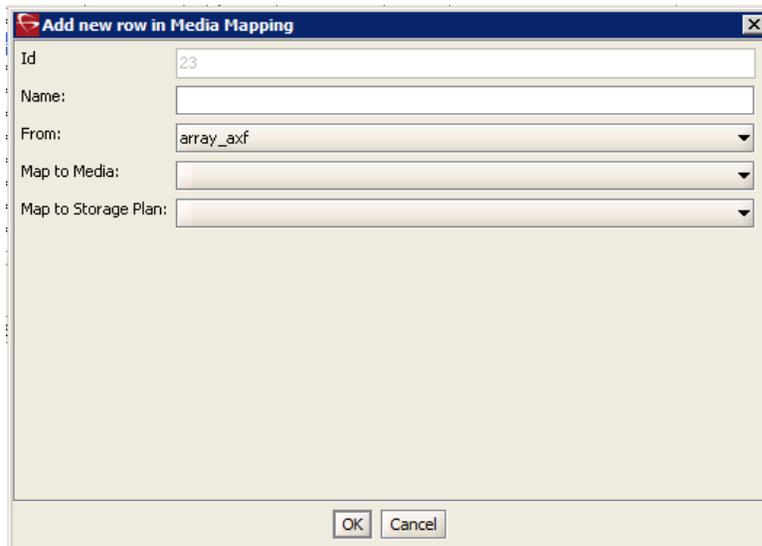
The example below demonstrates how the Format of a Tape Group can be changed. Changing Format of an Array is performed through the Array Editing windows in a similar manner.

Figure 38: Configuration Utility – Configuring the Format of a Tape Group



Click the “+” (*plus*) Button in the **Media Mapping Area** to add a new **Media Mapping** entry. Enter the **Name** for the new mapping. Then use the pull-down boxes to select the **Source (From)**, **Media to Map to** and/or **Storage Plan to Map to**.

Figure 39: Add New Row in Media Mapping Window



Click on the **Edit** Button in the **Media Mapping Area** to edit an existing **Media Map**.

Figure 40: Edit Media Mapping Entry Window

The screenshot shows a dialog box titled "Edit Media Mapping Entry". The dialog contains the following fields and values:

Field	Value
Id	24
Name:	demo
From:	array_axf
Map to Media:	array_axf
Map to Storage Plan:	Demo

At the bottom of the dialog are two buttons: "OK" and "Cancel".

2.15 DIVAprotect Tab

The DIVAprotect settings are identified within the Configuration Utility's **DIVAprotect Tab** as described in the following sections.

Figure 41: DIVAprotect Configuration Tab

The screenshot shows the Configuration Utility interface for the DIVAprotect tab. The window title is "Configuration Utility - DIVAprotect (User: diva1_70, connected to database on 127.0.0.1:1521)". The breadcrumb navigation is "System \ Robots \ Disks \ Drives \ Tapes \ Sets, Groups & Media Mapping \ **DIVAprotect** \ Media \ Storage Plans \ Slots \ Manager Setting \".

The interface is divided into three main sections:

- Configuration:** A table with two columns: "Parameter" and "Value".

Parameter	Value
DB: Maximum possible history of Events in Months	12
DB: Maximum possible number of Metrics	1,000,000
Manager: Size triggering Event Queue DB flush (nb events)	200
Manager: Time delay triggering Event Queue DB flush (secs)	15
- Event Definitions:** A list of event names with an "Open" button to the right.

Name
Analyze Error
Analyze
Archive Object
Checksum Verification Error Reading From Disk
Checksum Verification Error Reading From SD
Checksum Verification Error Reading From Tape
Copy As
Copy Instance
Create Instance
Delete Object
Delete Object Instance
- Metric Definitions:** A table with columns: "Enabled", "Name", "Description", "Resource", and "Collecting".

Enabled	Name	Description	Resource	Collecting
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE	Actor : amount of data READ and WRITTEN	Actor Name	Transfer Size
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_ABORTED_NUMBER	Actor : number of ABORTED READ and ABORTED WRITE operations with drives	Actor Name	Event ID
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_ABORTED_NUMBER_DAY	Actor : number of ABORTED READ and ABORTED WRITE operations with drives	Actor Name	Event ID
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_ABORTED_NUMBER_SD	Actor : number of ABORTED READ and ABORTED WRITE operations with SD	Actor Name	Event ID
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_ABORTED_NUMBER_SD_DAY	Actor : number of ABORTED READ and ABORTED WRITE operations with SD	Actor Name	Event ID
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_DAY	Actor : amount of data READ and WRITTEN	Actor Name	Transfer Size
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_NUMBER	Actor : number of READ and WRITE operations	Actor Name	Event ID
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_NUMBER_DAY	Actor : number of READ and WRITE operations	Actor Name	Event ID
<input checked="" type="checkbox"/>	ACTOR_TIME_ALL_OPERATION	Actor : time in all operations	Actor Name	Duration
<input checked="" type="checkbox"/>	ACTOR_TIME_ALL_OPERATION_DAY	Actor : time in all operations	Actor Name	Duration
<input checked="" type="checkbox"/>	ACTOR_TIME_READ	Actor : time in READ operations	Actor Name	Duration
<input checked="" type="checkbox"/>	ACTOR_TIME_READ_DAY	Actor : time in READ operations	Actor Name	Duration
<input checked="" type="checkbox"/>	ACTOR_TIME_WRITE	Actor : time in WRITE operations	Actor Name	Duration
<input checked="" type="checkbox"/>	ACTOR_TIME_WRITE_DAY	Actor : time in WRITE operations	Actor Name	Duration
<input checked="" type="checkbox"/>	ANALYZE_ABORTED_NUMBER	Analyzer : number ABORTED ANALYZER operations	Transcoder Name / Analyzer Name	Event ID

The main DIVAprotect configuration is done here.

The DIVAprotect Event Definitions and identified in this area.

DIVAprotect Metrics are defined in this area.

2.15.1 Configuration Area

The figure below depicts the **Configuration Area** of the DIVAprotect Tab:

Figure 42: DIVAprotect Tab Configuration Area Close-up

The screenshot shows a configuration window titled "Configuration" with a table of parameters and their values. Callouts provide detailed explanations for several parameters.

Parameter	Value
DB: Maximum possible history of Events in Months	12
DB: Maximum possible number of Metrics	1,000,000
Manager: Size triggering Event Queue DB flush (nb events)	100
Manager: Time delay triggering Event Queue DB flush (secs)	15

Callout 1: Enter the number of months to retain DIVAprotect Event History.

Callout 2: Enter the maximum number of DIVAprotect Metrics stored in the system. Once this number is exceeded, DIVAprotect will remove the oldest entries. This is done through an automated database job that executes once per day, every day.

Callout 3: This is the maximum interval for saving events to the database. If this interval is reached before the size triggering parameter is reached, the events will be saved to the database no matter how many have been collected.

Callout 4: Number of events collected in memory before saving them to the database.

2.15.2 Event Definitions Area

The figure below depicts the **Event Definitions Area** of the DIVAprotect Tab. The **Event Definitions** panel displays the list of Event Definitions available for use in the metrics. Event Definitions are factory set and can't be modified.

Figure 43: DIVAprotect Tab Event Definitions Area Close-up

The screenshot shows a panel titled "Event Definitions" with a list of event definitions and an "Open" button. A callout explains the function of the Open button.

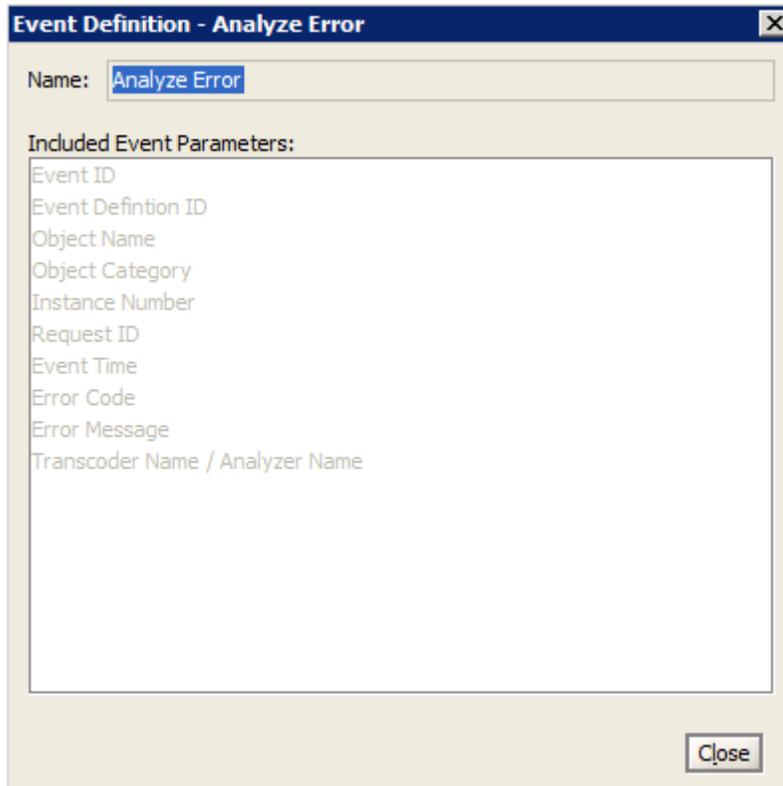
Name
Analyze Error
Analyze
Archive Object
Checksum Verification Error Reading From Disk
Checksum Verification Error Reading From SD
Checksum Verification Error Reading From Tape
Copy As
Copy Instance
Create Instance
Delete Object
Delete Object Instance

Callout: Highlight the desired Event Definition and click the Open Button to open the window in the next figure. This allows viewing the details of the selected Event Definition.

Note: Built-in metrics (DIVAPROTECT* metrics) can't be edited and therefore don't appear in the Metric Definitions panel.

Double-clicking on an Event Definition or clicking the **Open** Button will display a window listing its associated parameters as shown in the next figure.

Figure 44: Event Definition View Window



2.15.3 Metric Definitions Area

Double-clicking on a Metric Definition will display an edit dialog where the metric can be examined or modified. This has the same effect as selecting a metric in the list and clicking the *Edit* button.

The “+” (*plus*) and “-” (*minus*) buttons allow adding or deleting a metric.

Figure 45: DIVAprotect Tab Metric Definitions Area Close-up

Metric Definitions					+ Edit -
Enabled	Name	Description	Resource	Collecting	
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE	Actor : amount of data READ and WRITTEN	Actor Name	Transfer Size	▲
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_ABORTED_NUMBER	Actor : number of ABORTED READ and ABORTED WRITE operations with drives	Actor Name	Event ID	
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_ABORTED_NUMBER_DAY	Actor : number of ABORTED READ and ABORTED WRITE operations with drives	Actor Name	Event ID	
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_ABORTED_NUMBER_SD	Actor : number of ABORTED READ and ABORTED WRITE operations with SD	Actor Name	Event ID	
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_ABORTED_NUMBER_SD_DAY	Actor : number of ABORTED READ and ABORTED WRITE operations with SD	Actor Name	Event ID	
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_DAY	Actor : amount of data READ and WRITTEN	Actor Name	Transfer Size	
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_NUMBER	Actor : number of READ and WRITE operations	Actor Name	Event ID	
<input checked="" type="checkbox"/>	ACTOR_READ_WRITE_NUMBER_DAY	Actor : number of READ and WRITE operations	Actor Name	Event ID	
<input checked="" type="checkbox"/>	ACTOR_TIME_ALL_OPERATION	Actor : time in all operations	Actor Name	Duration	
<input checked="" type="checkbox"/>	ACTOR_TIME_ALL_OPERATION_DAY	Actor : time in all operations	Actor Name	Duration	
<input checked="" type="checkbox"/>	ACTOR_TIME_READ	Actor : time in READ operations	Actor Name	Duration	
<input checked="" type="checkbox"/>	ACTOR_TIME_READ_DAY	Actor : time in READ operations	Actor Name	Duration	
<input checked="" type="checkbox"/>	ACTOR_TIME_WRITE	Actor : time in WRITE operations	Actor Name	Duration	
<input checked="" type="checkbox"/>	ACTOR_TIME_WRITE_DAY	Actor : time in WRITE operations	Actor Name	Duration	
<input checked="" type="checkbox"/>	ANALYZE_ABORTED_NUMBER	Analyzer : number ABORTED ANALYZER operations	Transcoder Name / Analyzer Name	Event ID	▼

User: diva1_70, connected to database on 127.0.0.1:1521.

Click on the **Edit** Button in the **Metric Definitions Area** to edit an existing **Metric Definition**.

Figure 46: Metric Definition Editing Window

The screenshot shows a dialog box titled "Metric Definition - ACTOR_READ_WRITE_ABORTED_NUMBER". It contains the following fields and sections:

- Name:** ACTOR_READ_WRITE_ABORTED_NUMBER
- Description:** Actor : number of ABORTED READ and ABORTED WRITE operations with drives
- Enabled:**
- Collection:**
 - Collection Type:** Count (dropdown), Event ID (dropdown)
 - Weighted By:** Event ID (dropdown)
 - Collected Event:** A list of events with checkboxes:
 - Analyze Error
 - Analyze
 - Archive Object
 - Checksum Verification Error Reading From Disk
 - Checksum Verification Error Reading From SD
 - Checksum Verification Error Reading From Tape
 - Copy As
 - Copy Instance
- Aggregation:**
 - Resource:** Actor Name (dropdown)
 - Interval:** Lifetime (dropdown)

Buttons for **OK** and **Cancel** are located at the bottom right.

The **Description Field** allows the user to enter a description of the Metric Definition that will be shown next to the Metric Name in the Metric Definitions panel. This description also appears in the Control GUI when hovering with the mouse over an entry of the Metric Definition drop-down list.

The *Enabled* check box enables or disables data collection for the metric.

The *Collection Type* fields specify which event parameter (e.g. *Transfer Size*) is collected as the data and the statistical computation operated on it (e.g. *Sum*).

Available statistics are:

- Average
- Count
- Maximum
- Minimum
- Sum
- Weight-Based Average

The *Weighted By* field specifies the divider parameter for Weight-Based Average collection (e.g. *Duration*).

The *Collected Event* list specifies the events from which the collected event parameter is retrieved. The list will only display event types suitable for the parameter specified in the Collection Type second field. Event types that have no such parameter attached will be absent from the listing.

The *Resource Type* field specifies which resource is used to break down the data. For instance, if *Drive Serial Number* is selected, separate metrics will be generated for each drive.

The *Interval* specifies the interval for metric calculation. For instance, selecting “1 Day” will generate a metric each day (if corresponding data is available). The metric calculation will be based on the associated events that occurred in the last 24 hours.

2.15.4 Default Events and Metrics Configuration

The table below identifies the default events and metrics that are internal to the DIVArchive System.

Table 6: Default DIVArchive Event and Metric Definitions

Event Field ID	Displayed Name	Is Aggregatable? (Is Resource?)	Is Collectable?	Date / Number	Quantifier
1	Event Id	N	Y	N	--
2	Event Type	Y	N	--	--
3	Tape Type	Y	N	--	--
4	Tape Barcode	Y	N	--	--
5	Drive Type	Y	N	--	--
6	Drive Name	Y	N	--	--
7	Drive Serial Number	Y	N	--	--
8	Actor Name	Y	N	--	--
9	Object Name	Y	N	--	--
10	Object Category	Y	N	--	--
11	Object Instance	N	N	--	--
12	Media	Y	N	--	--
13	Request Id	N	N	--	--

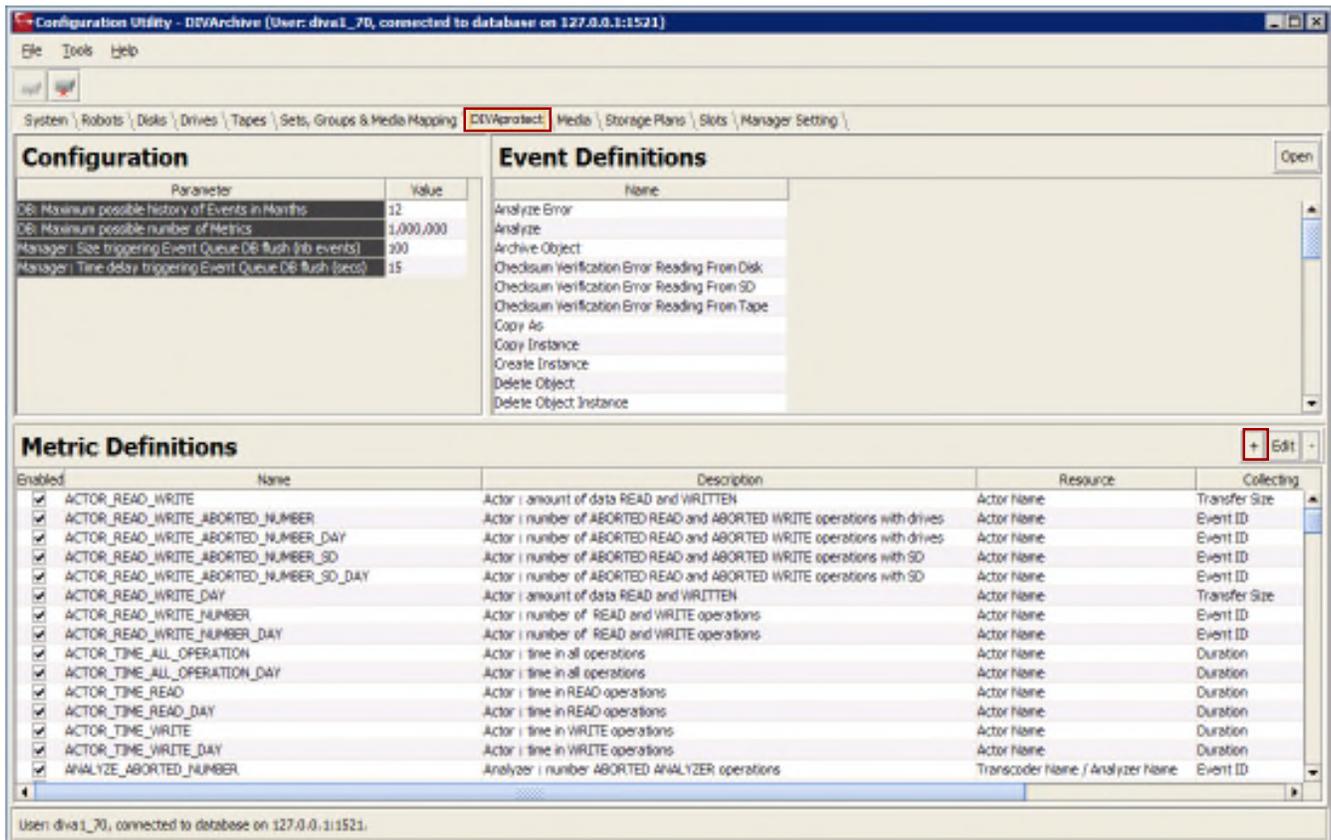
Event Field ID	Displayed Name	Is Aggregatable? (Is Resource?)	Is Collectable?	Date / Number	Quantifier
14	Event End Time	N	N	--	--
15	Event Duration	N	Y	N	Seconds
16	Transfer Size	N	Y	N	Bytes
17	Transfer Rate	N	Y	N	MB/Second
18	Transfer Error Rate	N	Y	N	Errors/GB
19	Error Code	Y	N	--	--
20	Error Message	N	N	--	--
21	Disk Name	Y	N	--	--
22	Library Serial Number	Y	N	--	--
23	SD Name	Y	N	--	--
24	Transcoder Name / Analyzer Name	Y	N	--	--
25	Local DIVArchive System	Y	N	--	--
26	Number of Operations	N	Y	N	--

2.15.5 Sample Metrics Definition

Use case: a user wants to create their own metric and is looking for average duration of read and write operations on a tape in a DIVArchive System.

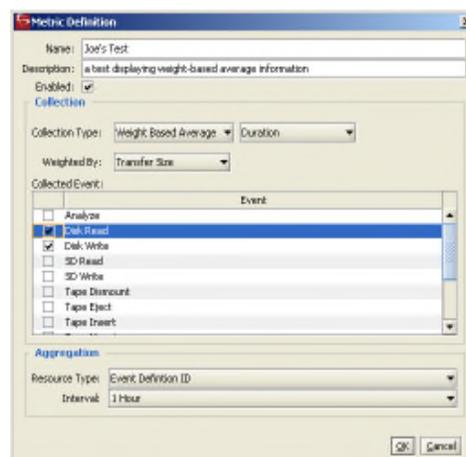
1. To create the metric go to the **DIVprotect Tab** of the Configuration Utility, locate the **Metric Definitions** pane, and click the “+” (*plus*) button:

Figure 47: DIVprotect Tab on the Configuration Utility and the “+” Button



2. The **Metric Definition** dialog window will open.

Figure 48: Metric Definition Window



3. Choose a unique name for your new metric. In this example it is set to “Joe’s Test”.
4. Add a description and enable it using the checkbox.
5. Set the **Collection Type** and **Weighted By** fields as appropriate.

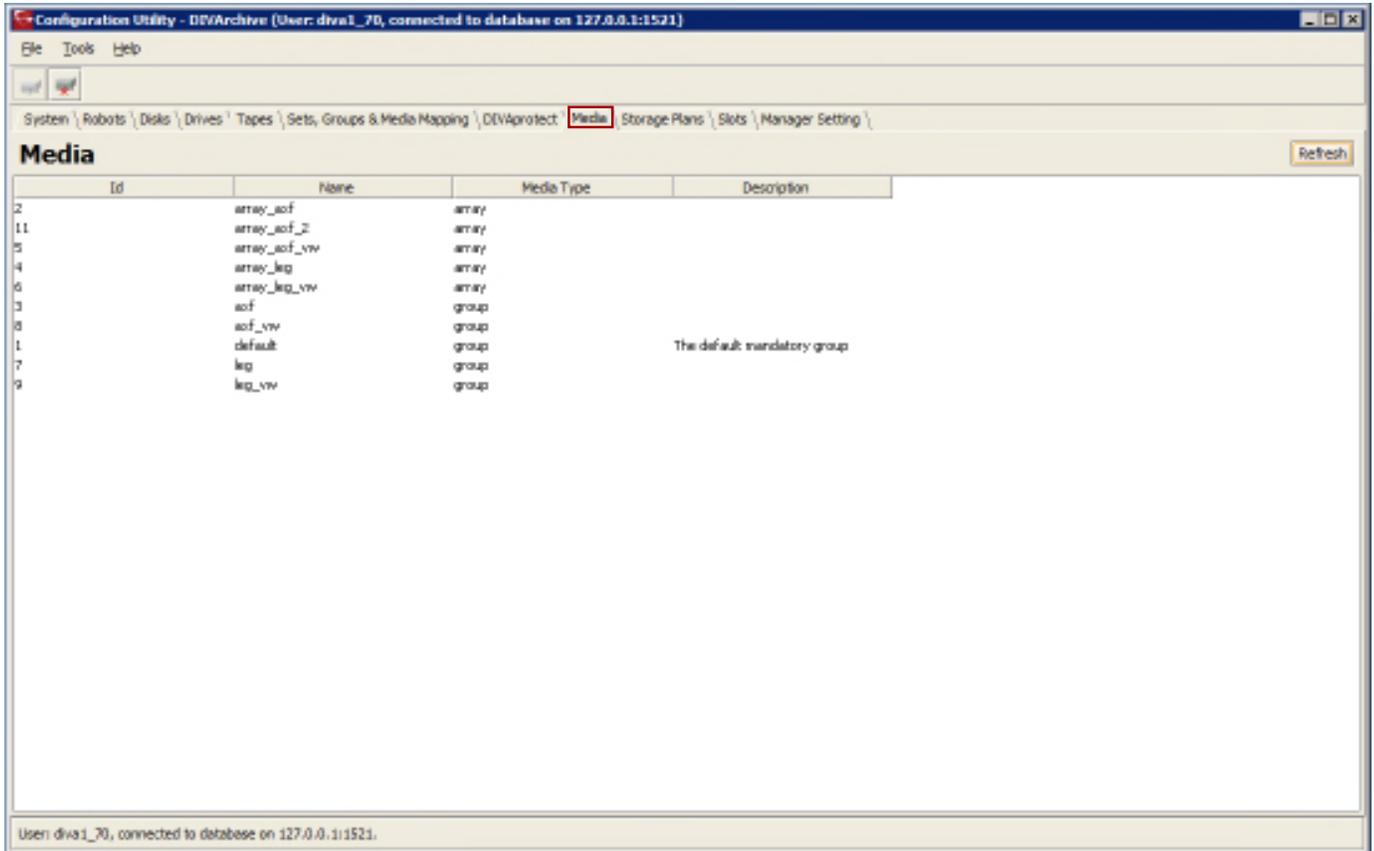
In the example the **Collection Type** chosen is **Weight-Based Average**. This **Collection Type** enables the **Weighted By** field and the user will be required to select a value to weight the **Metric Definition** by. The values for the **Weighted By** field will be identical to those in the second **Collection Type** field (*“Duration” in the example*).

6. Using the checkboxes, select the Events to collect.
7. Select the **Aggregation Resource Type**.
8. The **Aggregation Interval** can be left as the default (*one hour*).
9. Click the **OK** Button to complete the process.

2.16 Media Tab

The **Media Tab** displays information (*properties*) of the media identified in the DIVArchive System. The display is for informational purposes and read only. Use the **Refresh** Button to refresh the displayed list.

Figure 49: Media Tab Window



Id	Name	Media Type	Description
2	array_aof	array	
11	array_aof_2	array	
5	array_aof_vv	array	
4	array_leg	array	
6	array_leg_vv	array	
3	aof	group	
8	aof_vv	group	
1	default	group	The default mandatory group
7	leg	group	
9	leg_vv	group	

2.17 Storage Plans Tab

The **Storage Plans Tab** allows you to create simple and advanced rules for automated management and movement of material within the archive.

For detailed configuration information, refer to the *Oracle DIVArchive Storage Plan Manager (SPM) Guide* in the DIVArchive Additional Features library.

Figure 50: DIVArchive Storage Plans Tab Window

The screenshot shows the 'Storage Plans' tab in the DIVArchive Configuration Utility. The window title is 'Configuration Utility - DIVA' and the user is 'diva1_70, connected to database on 127.0.0.1:1521'. The 'Storage Plans' section contains a table with one entry: 'SP_DEFAULT' with 'Delete Last Instance' set to 'N', 'Origin' set to 'I', and 'Default Media' set to 'default'. The 'Media Groups' section is currently empty. The 'Filters' section is also empty. A callout box points to the 'Storage Plans' header, stating 'Storage Plan Name Definitions Frame.' Another callout box points to the 'Filters' section, stating 'Filter definitions relate to the Storage Plan Objects. It allows all or specific objects (based on object filters) to be actioned.' A third callout box points to the 'Media Groups' section, stating 'Defines the Tape Groups or Disk Arrays to be allocated to Slots, and if content deletion will be managed by the Storage Plan Manager.'

Name	Delete Last Instance	Origin	Default Media
SP_DEFAULT	N	I	default

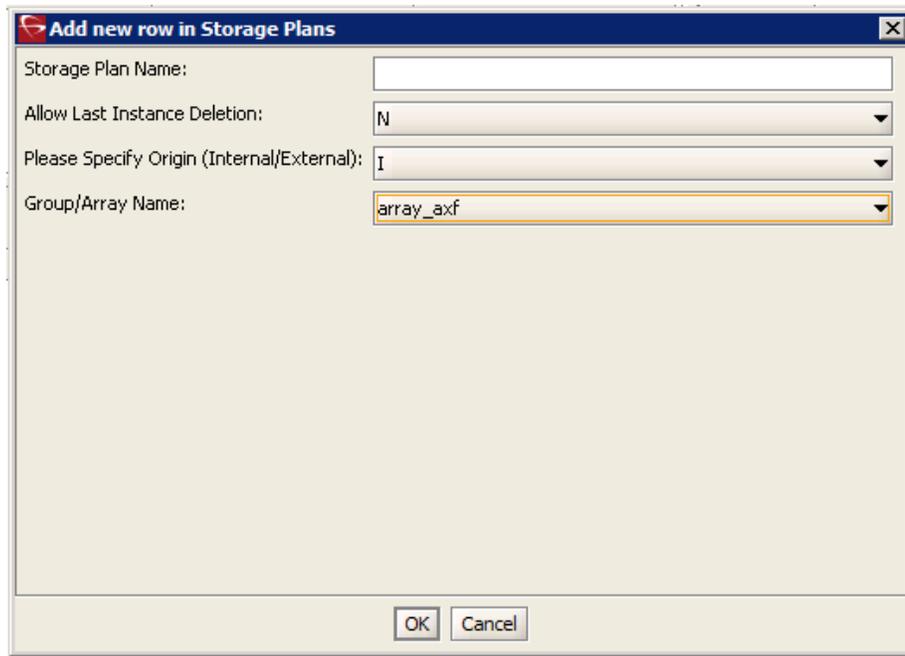
Name	Storage Name	Group/Array	Watermark	Cleaning
------	--------------	-------------	-----------	----------

Name	Filter Id	Media Filter	Category Filter	Source Filter	Name Filter	Min Size	Max Size	Storage Plan
------	-----------	--------------	-----------------	---------------	-------------	----------	----------	--------------

User: diva1_70, connected to database on 127.0.0.1:1521.

Click on the “+” (*plus*) Button in the **Storage Plans Area** to add a new **Storage Plan**. Enter the **Storage Plan Name** in the text box, then use the pull-down boxes to select whether or not to **Allow Last Instance Deletion**, **Internal or External Origin** (*typically Internal*), and the **Group/Array Name** to associate with the Storage Plan. Click the **OK** Button to save the changes.

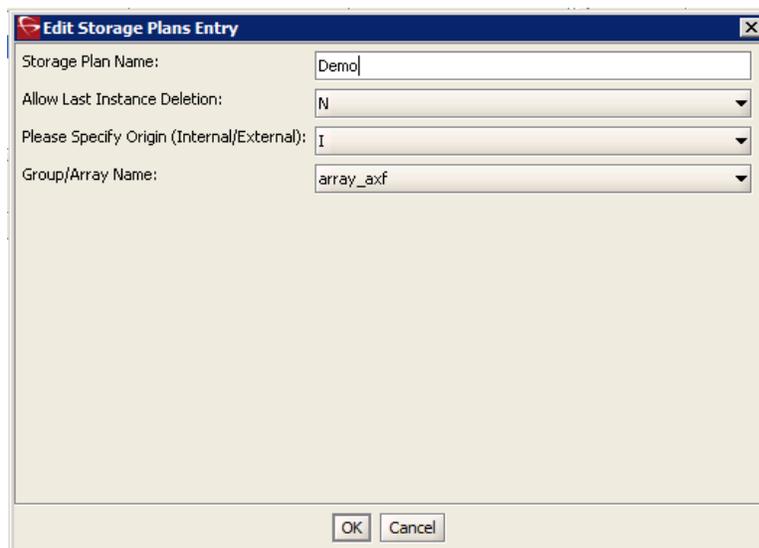
Figure 51: Add New Storage Plan Window



The screenshot shows a dialog box titled "Add new row in Storage Plans". It contains four input fields: "Storage Plan Name" (a text box), "Allow Last Instance Deletion" (a dropdown menu with "N" selected), "Please Specify Origin (Internal/External):" (a dropdown menu with "I" selected), and "Group/Array Name" (a dropdown menu with "array_axf" selected). At the bottom, there are "OK" and "Cancel" buttons.

Click on the **Edit** Button in the **Storage Plan Area** to edit an existing **Storage Plan**.

Figure 52: Edit Storage Plans Window



The screenshot shows a dialog box titled "Edit Storage Plans Entry". It contains four input fields: "Storage Plan Name" (a text box with "Demo" entered), "Allow Last Instance Deletion" (a dropdown menu with "N" selected), "Please Specify Origin (Internal/External):" (a dropdown menu with "I" selected), and "Group/Array Name" (a dropdown menu with "array_axf" selected). At the bottom, there are "OK" and "Cancel" buttons.

Click on the “+” (*plus*) Button in the **Media Groups Area** to add a new **Media Group**. Enter the **Medium Name** and **Storage Name** in the appropriate text boxes. Use the pull-down menus to select the **Group/Array Name**, whether or not to **Watermark** the medium, and the **Disk Cleaning Strategy**.

Figure 53: Add New Media Group Window

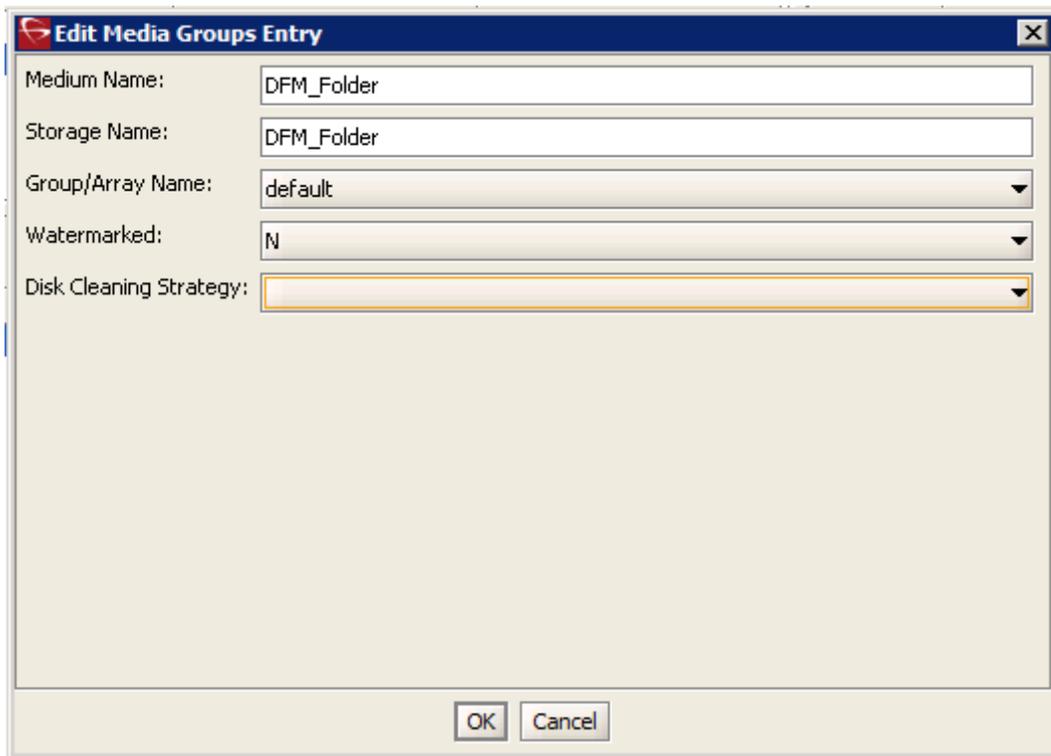
The screenshot shows a dialog box titled "Add new row in Media Groups". It contains the following fields:

- Medium Name: [Empty text box]
- Storage Name: [Text box containing "DIVA"]
- Group/Array Name: [Dropdown menu showing "array_axf"]
- Watermarked: [Dropdown menu showing "N"]
- Disk Cleaning Strategy: [Empty dropdown menu]

At the bottom of the dialog are "OK" and "Cancel" buttons.

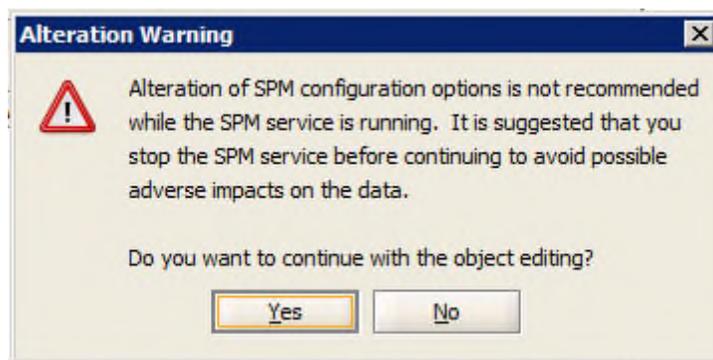
Click on the **Edit** Button in the **Media Groups Area** to edit an existing **Media Group**.

Figure 54: Edit Media Groups Entry Window



When all changes have been made to the Media Group, click the OK Button to save the changes. A warning window will appear asking if you want to continue with saving the changes. Click the **Yes** Button to save the changes, or the **No** Button to cancel.

Figure 55: Edit Media Groups Warning Window



2.18 Slots Tab

This tab defines the Slots associated with the Storage Plans for the Storage Plan Manager. Refer to the *Oracle DIVArchive Storage Plan Manager (SPM) Guide* in the DIVArchive Additional Features library for more details.

Figure 56: DIVArchive Slots Tab Window

Click the “+” (plus) Button to create a new Slot (see the figure below).

Click the Edit Button to edit an existing Slot (see the figure below).

Slots define which Tape Groups or Disk Arrays are related to each storage plan, and the parameters for storage plan execution.

Storage Plan Name	Slot Name	Request Type	Slot Begin	Slot End	Req. Execution Begin	Req. Execution End	Priority	Medium Name	Source	Req. Num of Ins	Tape Only	Enabled
Demo	Demo	STORAGE	10	20	24:00	07:00	30	demo_medium		1		Y

User: diva1_70, connected to database on 127.0.0.1:1521.

Click the “+” (*plus*) Button to create a new Slot. Configure the Slot’s parameters by entering the information desired for this slot, or using the pull-down boxes to select the options. Refer to the *Oracle DIVArchive Storage Plan Manager (SPM) Guide* in the DIVArchive Additional Features library for more information.

Figure 57: Slot Configuration Window

The screenshot shows a window titled "Slot Configuration" with a close button (X) in the top right corner. The window is divided into two main sections: "Slot Definition" and "Request Timings & Restrictions".

Slot Definition:

- Storage Plan*: A pull-down menu.
- Request Type: A pull-down menu with "STORAGE" selected.
- Slot Name*: A text input field.

Request Timings & Restrictions:

- Slot Begin Time(Minutes)*: A text input field.
- Request Execution Begin Time(HH:MM)*: A text input field with "00:00" entered.
- Slot End Time(Minutes)*: A text input field.
- Request Execution End Time(HH:MM)*: A text input field with "24:00" entered.
- Enabled*: A pull-down menu with "Y" selected.
- Request Priority*: A text input field with "30" entered.
- Medium Associated With Slot*: A pull-down menu.

Below these sections is a breadcrumb path: "Restore, Transcode & Metadata Archive \ Storage \".

At the bottom of the window are two buttons: "OK" and "Cancel".

Additional fields at the bottom of the window:

- Req. Number of Instances*: A text input field with "1" entered.
- Once Only*: A pull-down menu with "N" selected.

Note: The Slot Configuration window serves two purposes: new slot configuration as shown in the figure above, and editing an existing slot configuration as shown in the figure below. Both functions use the same window; however the information shown in the window is determined by whether a new slot is being added, or an existing slot is being edited.

Click on the **Edit** Button to edit an existing Slot. Refer to the *Oracle Storage Plan Manager (SPM) Guide* in the DIVArchive Additional Features library for more information.

Figure 58: Edit Slot Configuration Window

Slot Configuration

Slot Definition

Storage Plan* DFM

Request Type STORAGE

Slot Name* DFM_Slot

Request Timings & Restrictions

Slot Begin Time(Minutes)* 0 Request Execution Begin Time(HH:MM)* 00:00

Slot End Time(Minutes)* 0 Request Execution End Time(HH:MM)* 24:00

Enabled* Y Request Priority* 30

Medium Associated With Slot:* DFM_Folder

Restore, Transcode & Metadata Archive \ Storage \

Req. Number of Instances* 1

Once Only:* N

OK Cancel

2.19 Manager Setting Tab

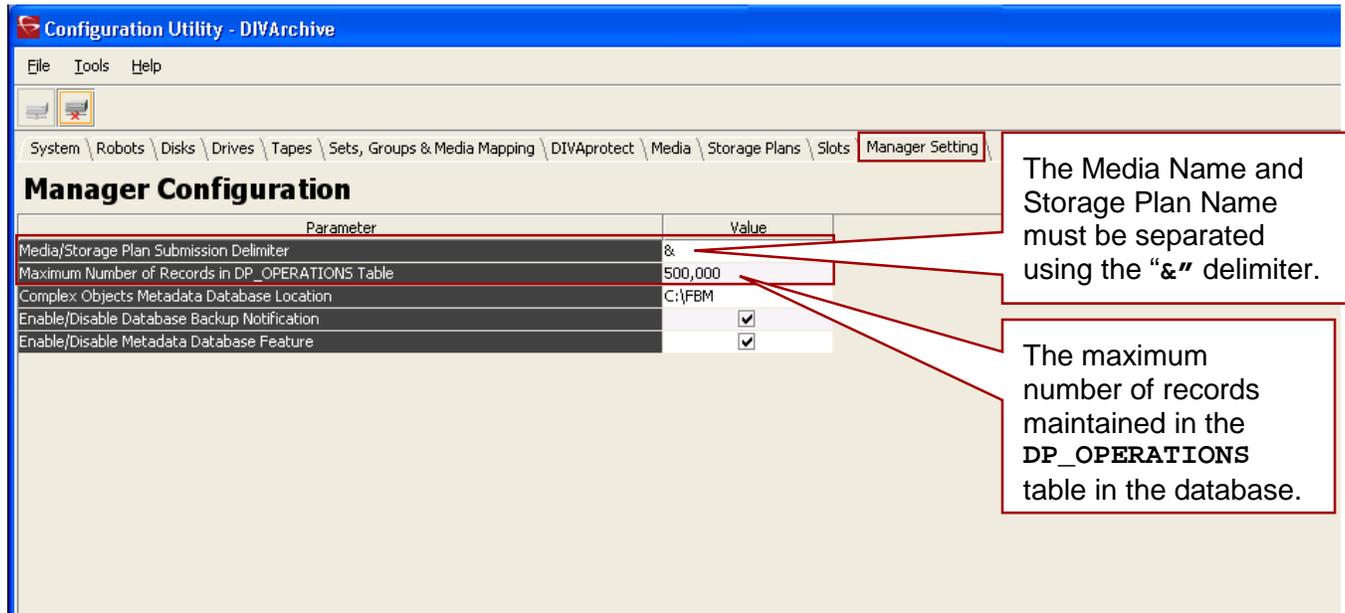
The **Manager Setting Tab** in the **Configuration Utility** is used to set several parameters related to the Media and the Metadata Database.

2.19.1 Media Configuration

There are two settings to configure for the Media in the Manager Setting Tab:

- Media/Storage Plan Submission Delimiter
 - The object is assigned to a specific Storage Plan and saved to the media specified in the SP. The Media Name and the SP Name must be separated by the delimiter “&”.
- Maximum Number of Records in **DP_OPERATIONS** Table
 - The maximum number of records maintained in the **DP_OPERATIONS** table in the database.

Figure 59: Configuration Utility Manager Settings Tab – Media Configuration



The screenshot shows the Configuration Utility interface with the Manager Setting tab selected. The window title is "Configuration Utility - DIVArchive". The breadcrumb path is "System \ Robots \ Disks \ Drives \ Tapes \ Sets, Groups & Media Mapping \ DIVAprotect \ Media \ Storage Plans \ Slots \ Manager Setting". The main area is titled "Manager Configuration" and contains a table with the following data:

Parameter	Value
Media/Storage Plan Submission Delimiter	&
Maximum Number of Records in DP_OPERATIONS Table	500,000
Complex Objects Metadata Database Location	C:\FBM
Enable/Disable Database Backup Notification	<input checked="" type="checkbox"/>
Enable/Disable Metadata Database Feature	<input checked="" type="checkbox"/>

Two callout boxes provide additional information:

- The first callout box points to the "Media/Storage Plan Submission Delimiter" value "&" and states: "The Media Name and Storage Plan Name must be separated using the “&” delimiter."
- The second callout box points to the "Maximum Number of Records in DP_OPERATIONS Table" value "500,000" and states: "The maximum number of records maintained in the **DP_OPERATIONS** table in the database."

2.19.2 Metadata Database Configuration

To enable Complex Objects processing, three parameters must be set in the Configuration Utility's Manager Settings Panel:

- **Enable/Disable Metadata Database Feature**
 - Check this to enable use of the Metadata Database.
 - Check the box to enable, uncheck to disable.
- **Complex Objects Metadata Database Location**
 - Set this to an existing empty directory in the file system.
- **Enable/Disable Database Backup Notification**
 - Check the box to enable, uncheck to disable.
 - Notifications must be enabled to receive DIVArchive Backup Service messages to the Control GUI. If this parameter remains disabled, there will be no notification of errors and/or warnings displayed in the Control GUI.

Figure 60: Configuration Utility Manager Settings Tab – Metadata Database

Parameter	Value
Media/Storage Plan Submission Delimiter	&
Maximum Number of Records in DP_OPERATIONS Table	500,000
Complex Objects Metadata Database Location	C:\YFBM
Enable/Disable Database Backup Notification	<input checked="" type="checkbox"/>
Enable/Disable Metadata Database Feature	<input type="checkbox"/>

3 Initial DIVArchive Configuration Settings

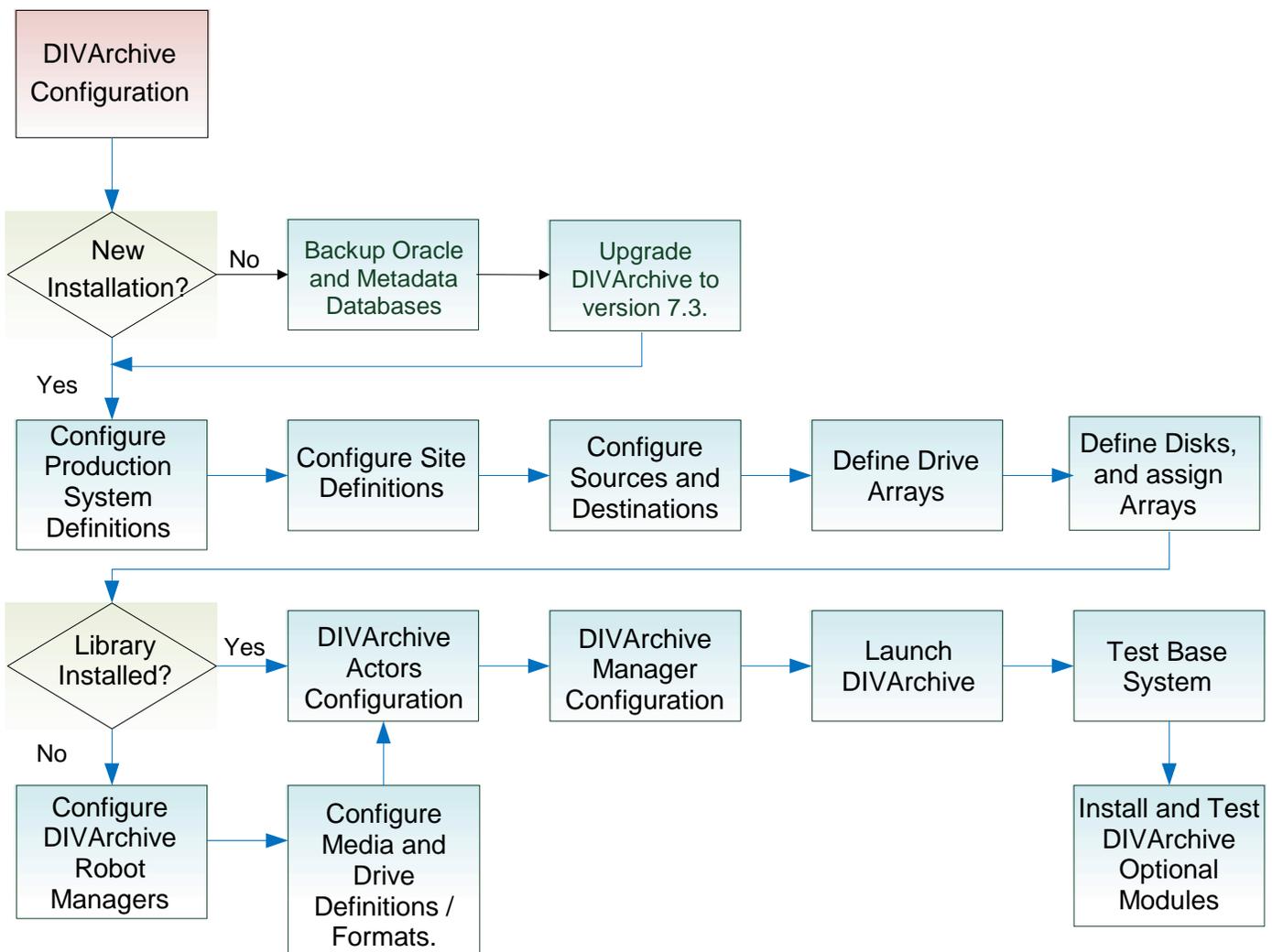
There are many inter-related components in a DIVArchive System. The basic workflow for configuration is shown in the following figure.

The configuration of DIVArchive is hierarchical and top-level parameters such as Production Systems, Sites, Arrays, and Disks need to be configured in advance of configuring other components such as DIVArchive Actors.

If you are intending to modify an existing DIVArchive System, you should always start by backing up the existing DIVArchive installation and configuration files, and above all else, **make a backup of the DIVArchive Oracle and Metadata Databases!**

If you are unsure about any steps in the procedures, or require clarification, you should contact Oracle Technical Support before making any modifications to your DIVArchive platform.

Figure 61: DIVArchive Configuration Overview



3.1 Production System and Sites Definitions

A **Production System** is a logical group within DIVArchive that associates Actors to your sources and destinations. It allows DIVArchive resources to be split between different applications or to prioritize particular functions over others within the platform by assigning more DIVArchive Actors to that Production System than the other. It should be noted that although DIVArchive Actors cannot be shared between Production Systems, it is possible to share a source and/or destination. In this case, the specific source or destination can be declared more than once **but** must have a unique name for each instance declared. **All installations must have at least one Production System defined.**

For applications that require extremely high bandwidth, the Production System concept also allows you to dedicate an Actor to an individual source.

For example, you may have a Production System used for on-air transmission, and a Production System for offline editing. The Production System concept allows you to fine-tune your allocation of resources between the two systems based on the workflow and bandwidth requirements of each of those two systems.

A **Site** allows an Actor Tape Library or Disk to be associated with a physical location. This allows DIVArchive to determine the most optimal use of resources during event execution. An example would be a remote installation connected over a WAN that is used for disaster recovery purposes for the primary site. **All installations have at least one Site Definition.**

Figure 62: Production System and Sites Definitions

The screenshot displays a software interface with two main sections: 'Prod. Systems' and 'Sites'. The 'Prod. Systems' section has a table with two rows: 'demo prod system 1' and 'demo prod system 2'. The 'Sites' section has a table with three rows: 'local', 'remote 1', and 'remote 2'. A callout box points to the '+' button in the 'Prod. Systems' section, stating 'Adds a new Production System.' Another callout box points to the '+' button in the 'Sites' section, stating 'Adds a new Site.' A third callout box points to the 'Is main site:' dropdown menu in the 'Add new row in Sites' dialog, stating 'Defines whether this Site should be considered first by the DIVArchive Manager for optimum resource allocation.' A fourth callout box points to the 'Storage Error' dialog box, stating 'Existing Production Systems or Sites that are already referenced by another DIVArchive Component (such as an Actor) are not removable until all references to it are removed.'

Prod. Systems

Prod.System Name
demo prod system 1
demo prod system 2

Sites

Site name	Site Id	Is main	Comments
local	1	Y	default site
remote 1	2	Y	demo remote ...
remote 2	3	Y	demo remote ...

Add new row in Prod. Systems

Production System Name: new system

OK Cancel

Add new row in Sites

Site Name: new site

Site Id: 4

Is main site: N

Comments: location

Cancel

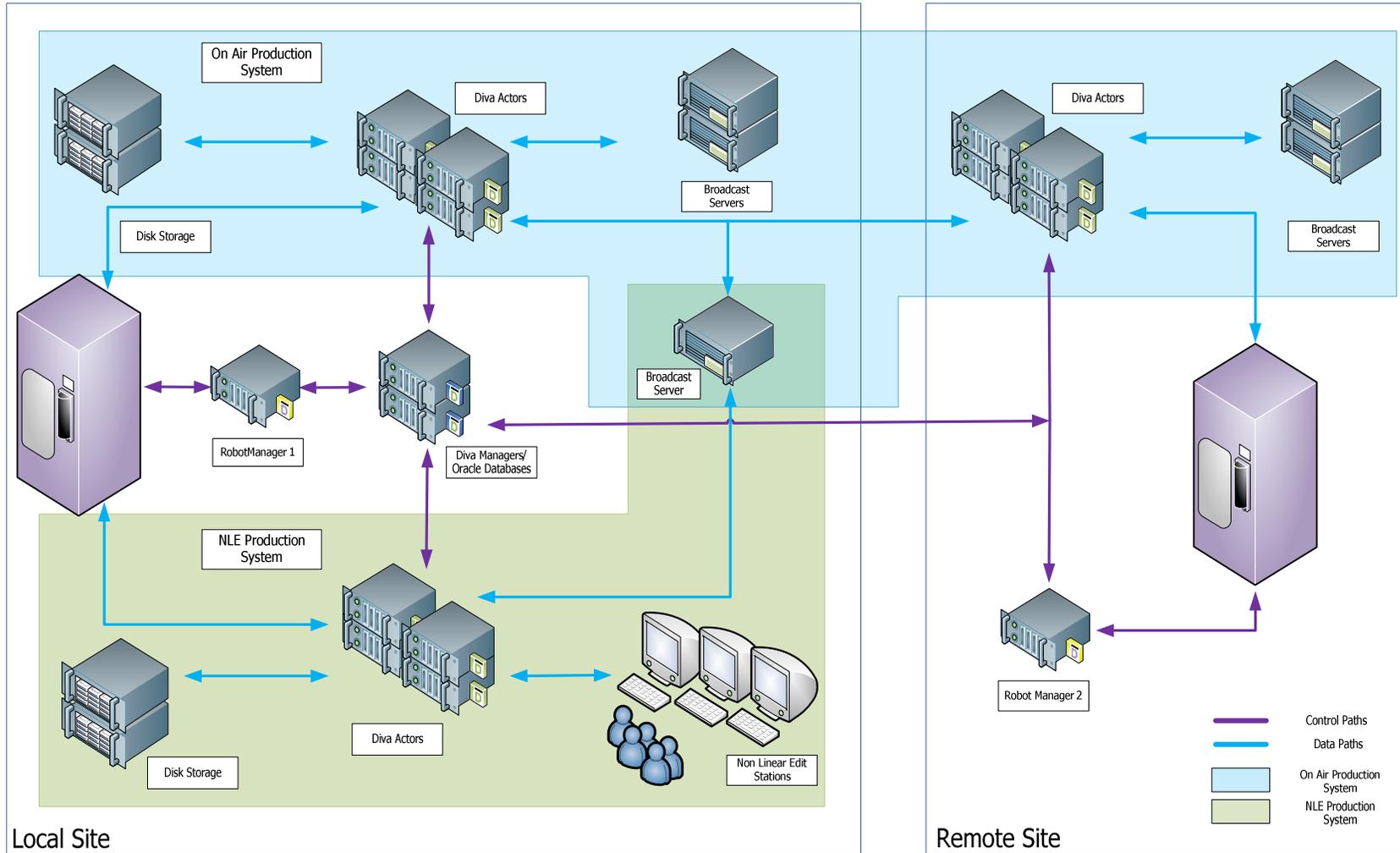
Storage Error

A storage error has occurred processing your request:
Error #2292: ORA-02292: integrity constraint (DIVA.SYS_C004087) violated - child record found

OK

Show/Hide detail...

Figure 63: Example of Production Systems and Sites

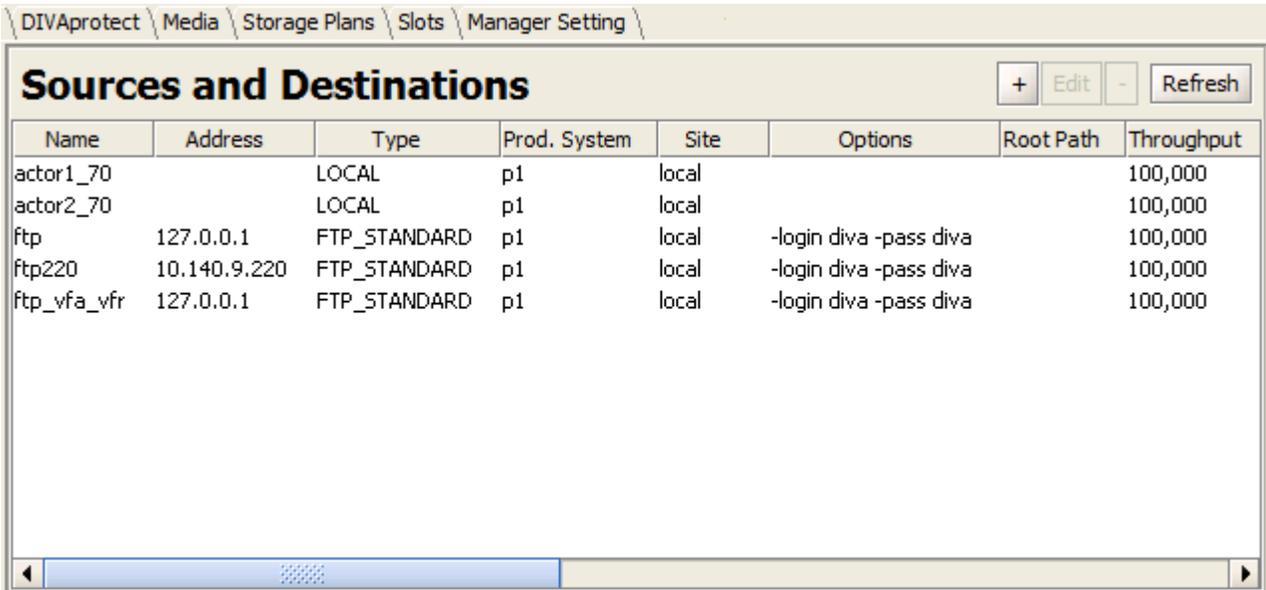


3.2 Sources and Destinations Definitions

A **Source** is defined as any connected system that contains content intended to be transferred to DIVArchive. A **Destination** is defined as any connected system that requires content to be transferred to it from DIVArchive. Examples of both are Broadcast Video Servers, FTP Servers, and Disk File Systems.

These entries are defined in the **Sources and Destinations** frame of the **System** tab in the Configuration Utility. Specific settings for each Source Type are described in Appendix C.

Figure 64: Source/Destination Area in the Configuration Utility System Tab



Name	Address	Type	Prod. System	Site	Options	Root Path	Throughput
actor1_70		LOCAL	p1	local			100,000
actor2_70		LOCAL	p1	local			100,000
ftp	127.0.0.1	FTP_STANDARD	p1	local	-login diva -pass diva		100,000
ftp220	10.140.9.220	FTP_STANDARD	p1	local	-login diva -pass diva		100,000
ftp_vfa_vfr	127.0.0.1	FTP_STANDARD	p1	local	-login diva -pass diva		100,000

Figure 65: Source and Destination Definitions

The screenshot shows the 'Edit Sources and Destinations Entry' dialog box with the following fields and callout boxes:

- Source Name:** actor2_70. Callout: Enter a Source Name. The Source Name must be unique. There can be multiple entries for the same physical source, as long as they have different names. This allows the source to be shared by multiple Production Systems.
- IP Address:** (empty). Callout: Enter the IP Address or Hostname of the source. Hostnames, if used, must be able to be resolved by the Actor Hosts. Some Source Types, such as **Disk**, do not require an IP Address and can be left blank for those Source Types.
- Source Type:** LOCAL. Callout: Select a Source Type from the drop-down menu. This defines the Source Type, which the Actor uses to select the correct communications protocol for the device.
- Prod. System:** p1. Callout: Specifies the Production System this source is dedicated to.
- Site:** local. Callout: Select the **Site** location to indicate the physical location of the Source.
- Connect Options:** (empty). Callout: Enter any parameters required to connect to the Source, such as username or password.
- Root Path:** (empty). Callout: Enter the root directory path (if any) for access to content on the Source.
- Max Throughput (Mbps):** 100000. Callout: This setting limits the total bandwidth used in transfers to or from this Source. This is useful when the Source does not natively provide any bandwidth throttling. This is typically set to a higher value than the source can provide (i.e. un-throttled).
- Max Accesses:** 10. Callout: Enter the maximum combined read and/or write operations permitted simultaneously to the source. For example, if this is set to 5, and there are 5 write requests active and additional read requests will be set to **Waiting for Resources** until at least one of the other requests have completed.
- Max Read Accesses:** 10. Callout: The **Maximum Read or Maximum Write accesses** allows for fine tuning of the transfers to the Source, such as more operations for restores than archives. If **Max Read Accesses** = 0 (Source only) If **Max Write Accesses** = 0 (Destination only).
- Max Write Accesses:** 10. Callout: Please see the Configuration Manual volume II, Section 5 for Checksum Support Configuration.
- External Checksum Source:** No (selected).
- Checksum Type:** (empty).
- GC Mode:** (empty).
- Verify Following Archive(VFA):** (unchecked).
- Verify Following Restore(VFR):** (unchecked).

3.3 Sources and Destinations Configuration Concepts

3.3.1 Source and Destination Configuration Limitations

If a Source/Destination is configured with the `-rxm1` option (typically used in DivaNet configurations to restore to a second DIVArchive System with checksums), a proper MD5 checksum is generated **only if the object contains a single file. If the object contains more than one file, no checksums are generated.**

When this is the case, and an object that contains more than one file is restored, the resultant `.xm1` Metadata File will **not** have checksums present, and a Source/Destination on the receiving side configured with GC active will fail to archive the object because the checksum verification will fail (no matching checksum).

3.3.2 Source Type

The **Source Type** parameter of the Source or Destination Definition establishes the specific protocol to be used for the interface between the DIVArchive Actor and the target device or file system.

The Source Types supported by DIVArchive are predefined and selection is limited to the listing from the Source Type Drop-down menu as shown below. Additional Source Types may be added in future versions of DIVArchive as new interfaces are introduced.

Note: *Deprecated* indicates that this is a software feature or function within DIVArchive that has subsequently been replaced by an improved feature set, however it is still currently supported by DIVArchive.

Table 7: Description of Source Types

Source Type	Description
AM Communicator	AVID Archive Manager Interface
TM Communicator DET Interface	AVID Unity interface using the Dynamically Extensible Transfer (<i>DET</i>) protocol.
TM Communicator DHM Interface	AVID Unity interface using the Data Handling Module (<i>DHM</i>) protocol.
CIFS	A Disk Source that is assumed to be visible to all Actors in the associated Production System. Buffered I/O will be used for transfers to/from this source.
DISK	A Disk Source that is assumed to be visible to all Actors in the associated Production System. Direct I/O will be used for transfers to/from this source.
EXPEDAT	DataExpedition (Expedat) Server
FTP_STANDARD	FTP protocol for RFC959 compliant servers.

Source Type	Description
LEITCH	This source type is deprecated, and is only provided for backward compatibility for older Leitch Servers. More recent servers should use the FTP_STANDARD source type.
LOCAL	Used to represent a disk partition bound to a specific Actor.
MEDIAGRID	Omneon MediaGrid.
METASOURCE	Used for dealing with multiple servers sharing the same online storage.
MSS	AVID (<i>formerly Pinnacle</i>) MediaStream Servers
OMNEON	Omneon Spectrum Servers utilizing unique FTP site commands.
PDR	GVG Profile Servers
QUANTEL_ISA	Quantel Q or sQ Servers utilizing the Quantel Power Portal.
QUANTEL_QCP	Older Quantel Servers utilizing the QCP protocol.
SEACHANGE_BMC	SeaChange Broadcast Media Clusters or Media Servers using Vstream Streaming API.
SEACHANGE_BML	SeaChange Broadcast Media Libraries using FTP or CIFS protocols.
SEACHANGE_FTP	SeaChange FTP support; this Source Type is deprecated.
SFTP	Secure Shell (<i>SSH</i>) FTP protocol.
SONY_HYPER_AGENT	For use with Newbase's FTP server by Sony.
Simulation	For Simulator platforms only, this creates a simulated Source/Destination.

One record is required for each ContentDirector DIVArchive has to move data from/to.

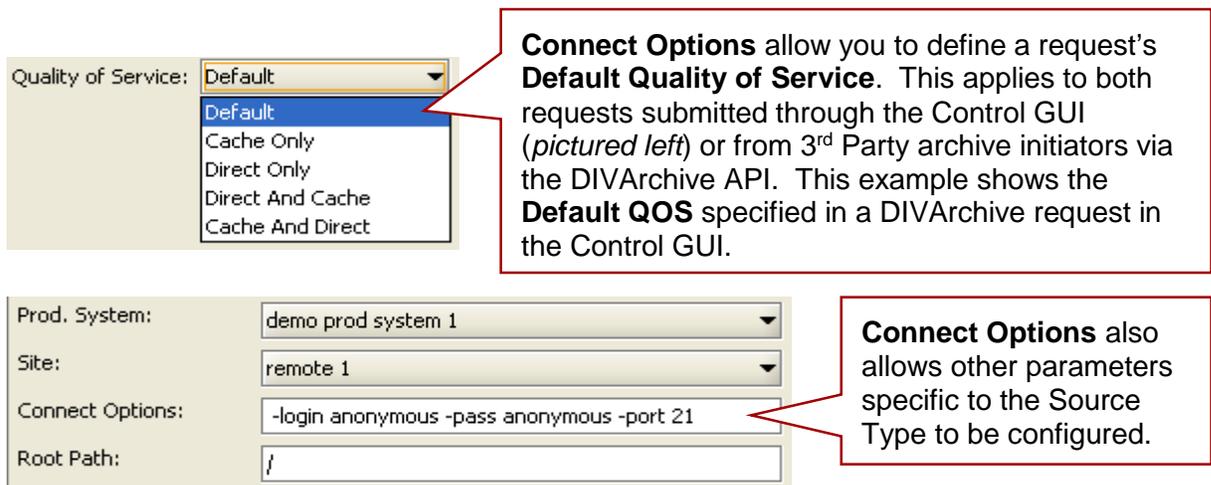
Table 8: ContentDirector Attributes

Attribute	Value	Example
IP Address	To be left empty	
Source Type	MEDIAGRID	MEDIAGRID
Root Path	\\ContentDirector\filesystem\clip.dir	\\10.30.0.200\cdev4\clip.dir \\mycontentdir\fs5\clip.dir

3.3.3 Connect Options

The **Connect Options** box allows you to define the **Default Quality Of Service** to be used in DIVArchive transfer requests and/or additional protocol specific parameters (*such as username and password*) to allow the DIVArchive Actor to establish a connection to the target device or disk file system.

Figure 66: Connection Options



Specific connect options for the supported Broadcast Servers or file system types are covered in detail in the appendix.

3.3.4 Root Path

Although the directory path can be explicitly specified at the request level, the **Root Path** box allows you to define the default directory path for the source, or as a disk mount point for disk and local sources. The Root Path specified will be appended before any Files Path Root specified in an archive, restore, or Partial File Restore request, unless the Files Path Root specified in the request is an absolute path.

The benefit of the Root Path approach is that it allows the server's directories to be specified at the source and destination configuration level, rather than at the request level, and can be altered without affecting commands that are issued from DIVArchive Clients.

The interaction of the Files Path Root and Root Path parameters are covered in detail in the Appendix.

3.3.5 Metasource (Option)

Metasource provides the ability for DIVArchive to combine two or more existing Source/Destinations, which use common storage, or multiple Drop Folder Monitors (DFMs), into a single Source/Destination configuration.

This concept allows load balancing and fault tolerance from within DIVArchive in case of an individual server or DFM failure. If the attached server or DFM is unavailable in a request, the next server or DFM in the Metasource Configuration is automatically tried.

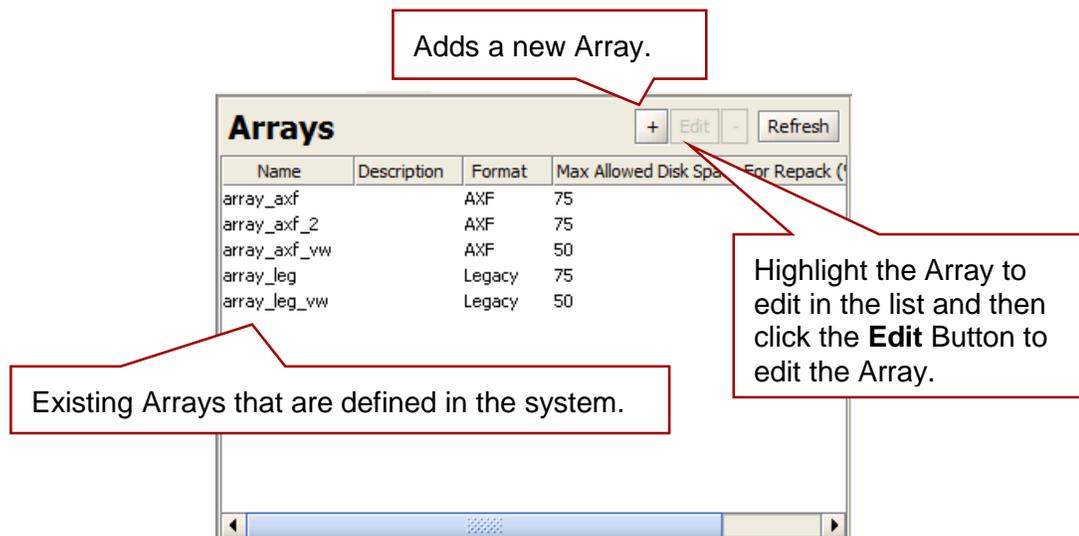
3.4 Arrays and Disks

DIVArchive's approach to disk management defines each physical disk, defines how it is attached (or mounted) to the system, and then groups the disks together to perform specific roles within the archive.

3.4.1 Defining an Array

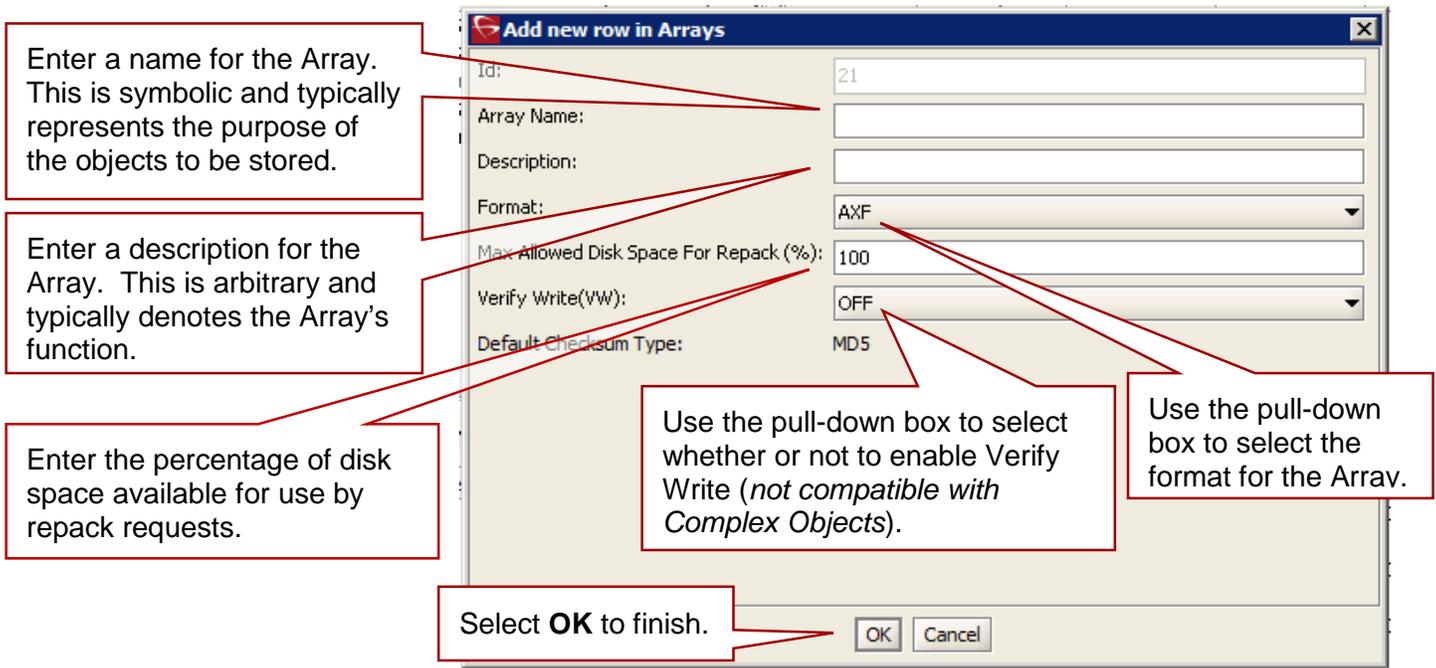
The first step to disk management is to define an **Array**. In DIVArchive an Array is a logical grouping of one or more disks for the storage of DIVArchive Objects.

Figure 67: Disks Tab – Arrays Area Close-up



Click on the “+” (*plus*) Button to add and define a new Array. Fill out the necessary information in the text box areas, then using the pull-down menus select the Array's **Format** and whether or not to turn on **Verify Write**. The Array's ID is automatically generated by the system and is not editable. Click the **OK Button** to save the new Array.

Figure 68: Add New Array Window



Click on the **Edit Button** to edit an existing Array. When finished editing, click the **OK Button** to save the changes.

Figure 69: Edit Array Window

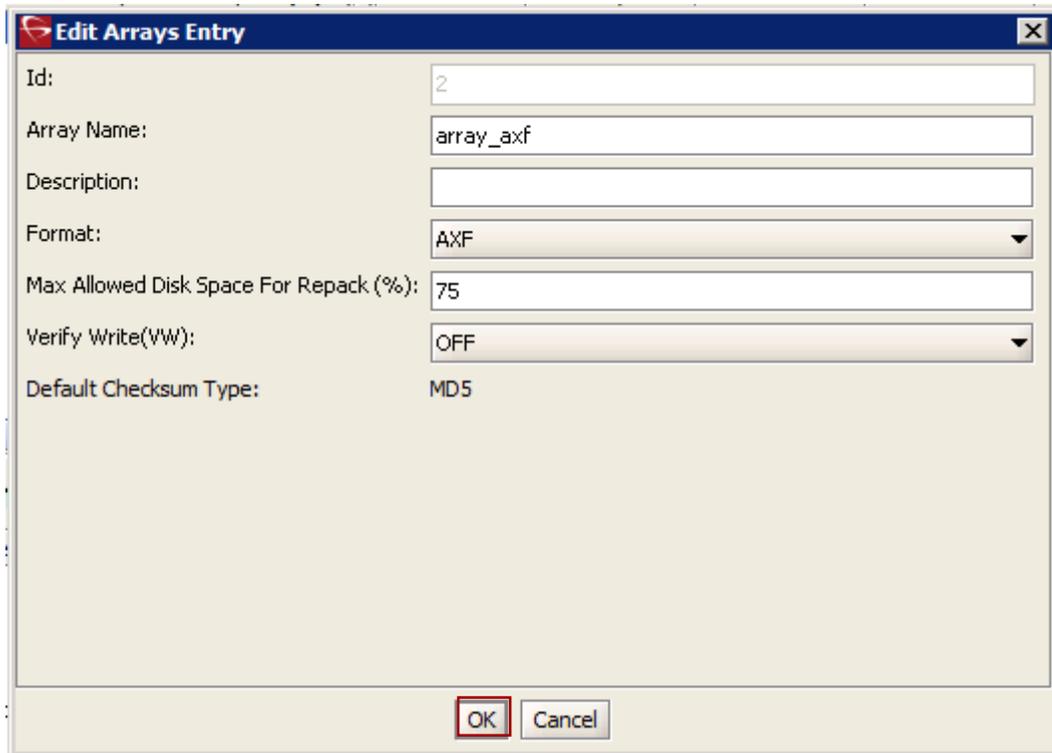
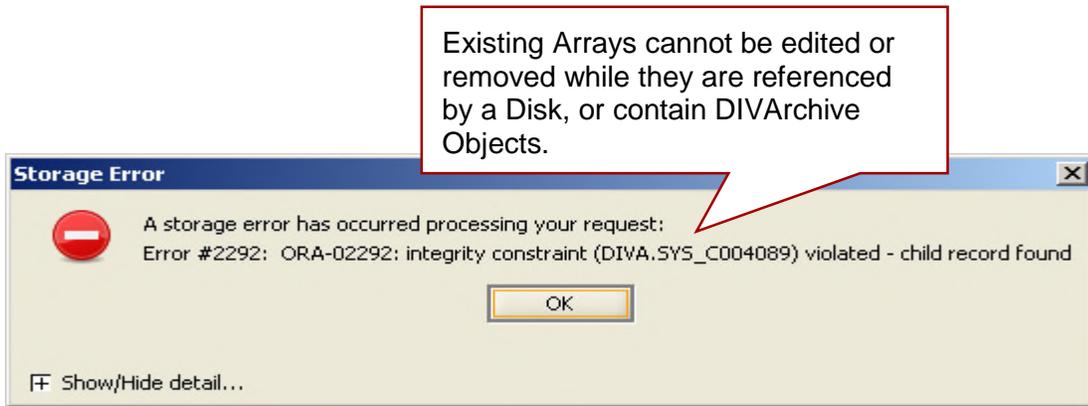


Figure 70: Editing an Array Warning Message



3.4.2 Defining Disks

The next step is to define the physical disks that are going to be used by DIVArchive, and assign them to Arrays based on their intended function.

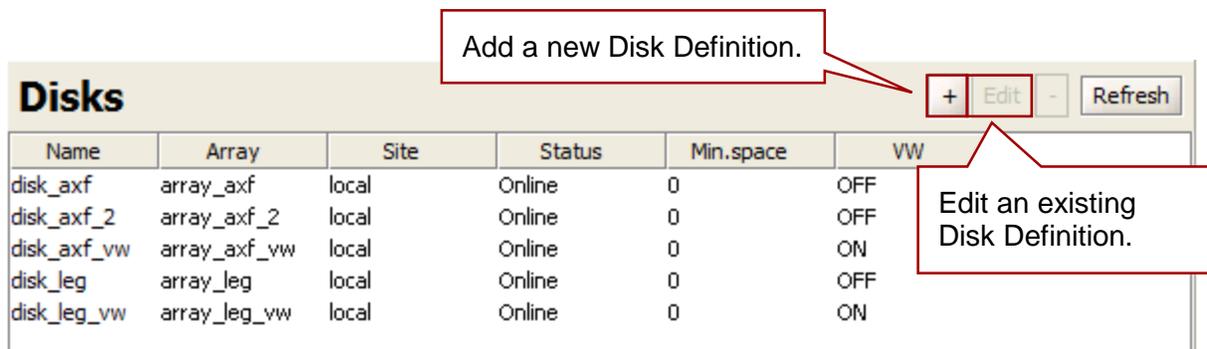
Each disk that is configured in the **Disks Frame** represents a distinct **physical** volume.

Note: Logical associations of disks to DIVArchive are performed in the Actor-Disk Connections Frame.

Each disk declaration in this frame can be assigned to one Array *only*. If a physical disk is intended to be shared between two or more Arrays, the disk can be declared multiple times but each declaration must have a unique name.

Defining how the disks are actually interfaced to DIVArchive is performed in the **Actor-Disk Connections** frame of the **Disks** tab of the Configuration Utility.

Figure 71: Disk Tab – Disks Area Close-up



Click on the “+” (*plus*) Button to add a new Disk Definition to the system.

Figure 72: Add New Disk Window

The screenshot shows the 'Add new row in Disks' window with the following fields and callouts:

- Disk Name:** A text input field. Callout: "Enter a symbolic name for the disk. This could describe its function or its location."
- Array:** A dropdown menu showing 'array_axf'. Callout: "Select the Array this disk is to be a member of. Only arrays currently configured in the Arrays Frame are listed in the drop-down menu."
- Site:** A dropdown menu showing 'local'. Callout: "Select the Site that defines the location of this disk. If Site Selection is enabled in the DIVArchive Manager Configuration File; this parameter is considered by DIVArchive for the optimum allocation of disk resources in the Array for object storage or retrieval."
- Status:** A dropdown menu showing 'ONLINE'. Callout: "Sets the current status of the disk. Offline indicates that the disk is offline and not to be used. During DIVArchive operations, the status may be set Offline by DIVArchive in the case of an unexpected I/O error with the disk."
- Min. Free Space, MB:** A text input field showing '0'. Callout: "Set the minimum free space of the disk. When the remaining free space reaches this amount, DIVArchive considers the disk full. Use this setting on disks that are shared with other applications, or with file systems that suffer poor or degraded performance when approaching 100% capacity."
- Verify Write(VW):** A dropdown menu. Callout: "Verify Write(VW) will be selected from Arrays if disk has none selected."
- Default Checksum Type:** A text input field showing 'MD5'. Callout: "Please see the Configuration Manual volume II, Section 5 for Checksum Support Configuration."

Buttons: OK, Cancel

Click the **Edit** Button to edit an existing Disk. Make the desired updates and click the **OK** Button to save the changes.

Figure 73: Edit an Existing Disk Window

The screenshot shows the 'Edit Disks Entry' window with the following fields:

- Disk Name:** disk_axf
- Array:** array_axf
- Site:** local
- Status:** ONLINE
- Min. Free Space, MB:** 0
- Verify Write(VW):** ...
- Default Checksum Type:** MD5

Buttons: OK, Cancel

4 Robot Manager Configuration

The DIVArchive Robot Manager on Windows platforms runs as a Windows Service and is launched automatically with Windows (*since DIVArchive version 6.0*).

The type of interface to be used for a specific library is configured in a static configuration file with the filename `RobotManager.conf` and is located in the `DIVA_HOME\Program\conf\robot_manager` folder on the host where the DIVArchive Robot Manager software has been installed. In a new installation (*or upgrade*) the file is provided with an `.ini` extension, which must be removed for it to be acknowledged by the Robot Manager. The normal procedure is to copy the file, remove the `.ini` extension, and then edit the new file.

Some settings in the configuration file are initially commented out (*i.e. they have a hash (#) in front of the parameter*). This indicates to the Robot Manager that the setting is ignored. For the setting to be taken into account the comment (`#`) must be removed.

Since many different types of libraries and connections are supported, not all sections of the configuration file will be relevant to your particular installation. Additionally, some parameters are specific to the operating system that the Robot Manager is installed upon (*i.e. Windows*).

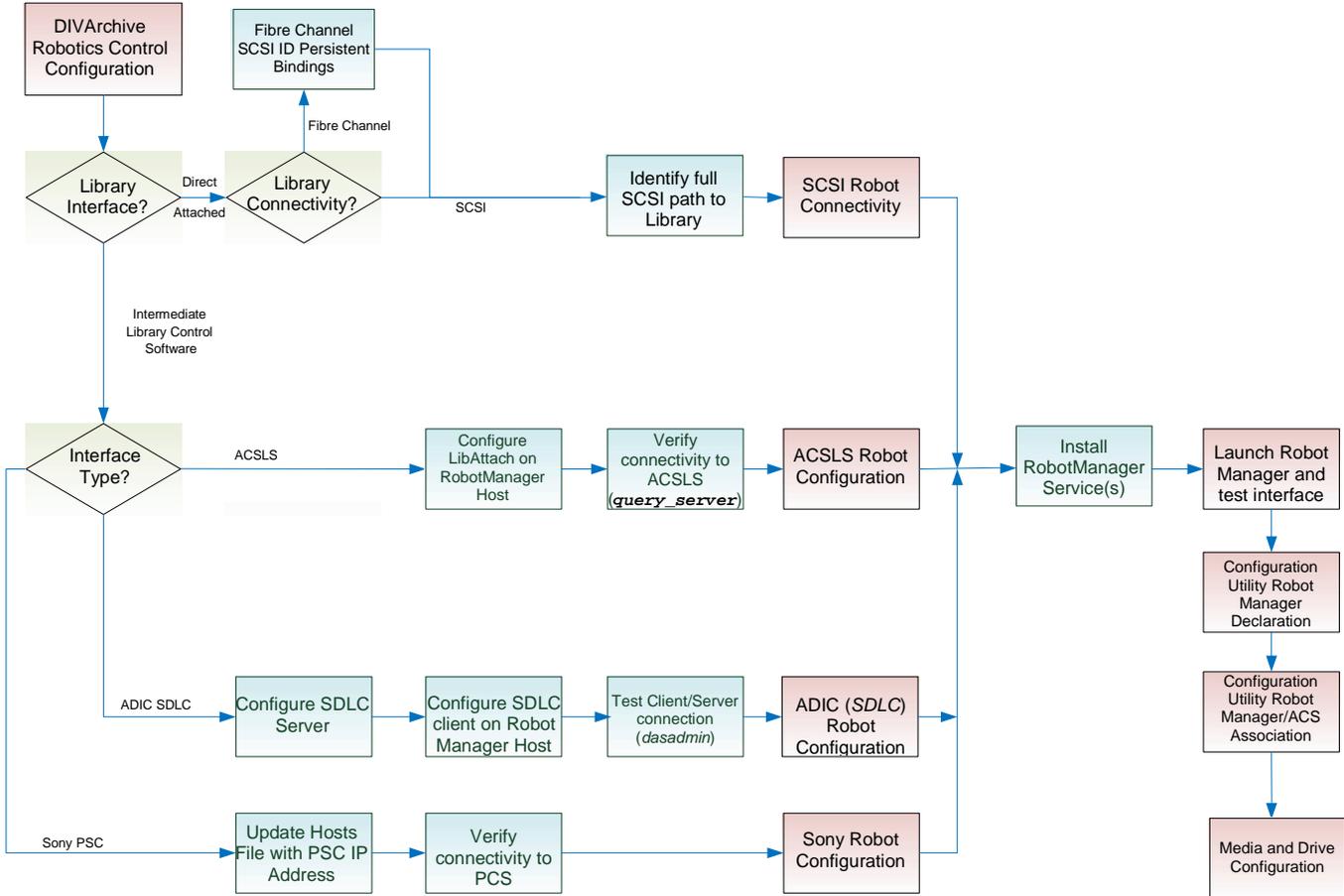
LocalDelete now has two modes of operation:

- The default legacy behavior called *ASAP Purge Mode*.
 - This mode purges objects immediately after they are copied to other sites.
- The new *Delayed Purge Mode*.
 - This mode operates with watermarks in a method similar to SPM. For more information please refer to the conf files LocalDelete configuration file folder.

4.1 Configuration Overview

The following figure outlines the steps for configuring the robotics for control by DIVArchive.

Figure 74: Configuration Overview



4.2 SCSI Connected Libraries

For directly attached SCSI controlled libraries, you must configure and correctly identify the SCSI ID controlling the library, and enter this value into the `RM_SCSI_DEVICE_LSM` parameter in the Robot Manager Configuration file. Before this can be done, a few concepts should be understood as described in the following sections.

Table 9: `robotmanager.conf` Parameters

Parameter	Description
<code>RM_SCSI_MOVEMEDIUM_TIMEOUT</code>	<p><code>MOVE MEDIUM</code> is the SCSI command use by Robot SCSI during mount, dismount, enter, and eject requests. The default timeout is 15 minutes for the communication between the Library and the RobotManager.</p> <p>Some libraries like Spectra T950 may require more time to be able to complete a <code>MOVE MEDIUM</code> request.</p> <p>The value of <code>RM_SCSI_MOVEMEDIUM_TIMEOUT</code> is indicated in minutes.</p>
<code>RM_SCSI_EJECT_USEGLOBALLOCK</code>	<p>This parameter should be set to 1 if you want to have the SCSI RobotManager Eject Calls obtain the lock for number of the LSM, and hold that lock until all associated tapes to be ejected have completed the ejection process; then unlock the drive and move onto the next drive. The default setting is 0.</p> <p style="text-align: center;"><code>RM_SCSI_EJECT_USEGLOBALLOCK=1</code></p>

4.2.1 Fiber Channel HBA's and SCSI Persistent Binding

Most installations now use Fiber Channel (FC) rather than native SCSI to interface to the library (*typically over a SAN*). In these instances, the FC Host Bus Adaptor (HBA) in the DIVArchive Robot Manager host presents the World Wide Name of the library interface as a SCSI ID. By default, most HBA's will automatically map these to a SCSI ID for the host operating system to access. This however presents a problem if a device is added or removed on the SAN as this may alter the SCSI ID of the library by the HBA, automatically re-mapping the existing devices. To avoid this problem disable the **Automap** feature and use **Persistent Bindings** instead. This feature allows the SCSI mapping of the library to remain consistent between reboots of the host, and from the advent of any addition or removal of devices on the SAN.

Below is an example of a WWPN to SCSI ID Persistent Binding for an Emulex Host Bus Adaptor. For detailed instructions on enabling persistent bindings for your particular HBA, refer to the manufacturer's installation manual.

Figure 75: Example of WWPN to SCSI ID Binding

The screenshot shows the HBAAnyware Utility interface. On the left, a tree view under 'Discovered Elements' shows a 'DIVAMGR' device with three WWPNs: 50:01:04:F0:00:82:25:98, 50:01:04:F0:00:82:25:A2, and 50:01:04:F0:00:82:25:A5. The main panel has tabs for 'General', 'Details', 'Port Attributes', 'Port Statistics', 'Firmware', 'Target Mapping', and 'Driver Parameters'. The 'Target Mapping' tab is active, showing 'Current Settings' with 'Active Bind Type: WWPN' and 'Automapping: Disabled'. Below this is a 'Current Mappings' table:

wwPN	wwNN	D_ID	SCSI ID	Type
50:01:04:F0:00:82:25:98	50:01:04:F0:00:82:25:97	10200	(0 , 0)	PB
50:01:04:F0:00:82:25:A2	50:01:04:F0:00:82:25:A1	10326	(0 , 1)	PB
50:01:04:F0:00:82:25:A5	50:01:04:F0:00:82:25:A4	10426	(0 , 2)	PB

Below the table is a 'Persistent Binding Configuration' section with a table:

Target WWPN	SCSI ID
50:01:04:F0:00:82:25:98	(0 , 0)
50:01:04:F0:00:82:25:A2	(0 , 1)
50:01:04:F0:00:82:25:A5	(0 , 2)

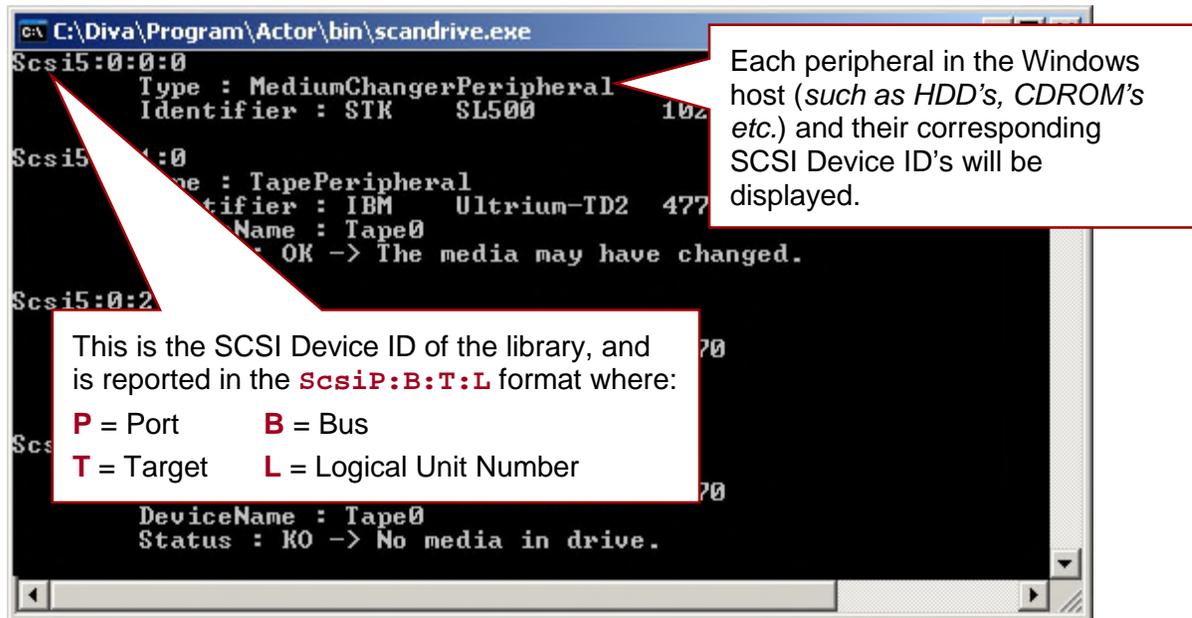
To the right of this table is a 'Display Mode' section with radio buttons for 'Show WWPN' (selected), 'Show WWNN', and 'Show D_ID'. At the bottom are buttons for 'Add Binding', 'Bind New', 'Remove', and 'Remove All'. A red callout box points to the WWPN column in the 'Current Mappings' table with the text: 'The WWPN of the library can usually be found through the front control panel interface of most libraries.'

It is also important to mention that if the library controller or the HBA in the DIVArchive Robot Manager host is changed, this may alter the library's SCSI Persistent Bindings to the host operating system. This will require the Persistent Binding for the library to be reconfigured in the HBA configuration software on the DIVArchive Robot Manager host.

4.2.2 Determining the SCSI Library Connection

For Windows, the `RM_SCSI_DEVICE_LSM(x)` settings for the DIVArchive Robot Manager can be determined by using the `scandrive.exe` utility; this can be obtained from the `DIVA_HOME\Actor\bin` directory. This utility automatically reports all devices located in the Windows SCSI hardware tree in the registry and their corresponding port, bus, target, and Logical Unit Numbers (*LUN*).

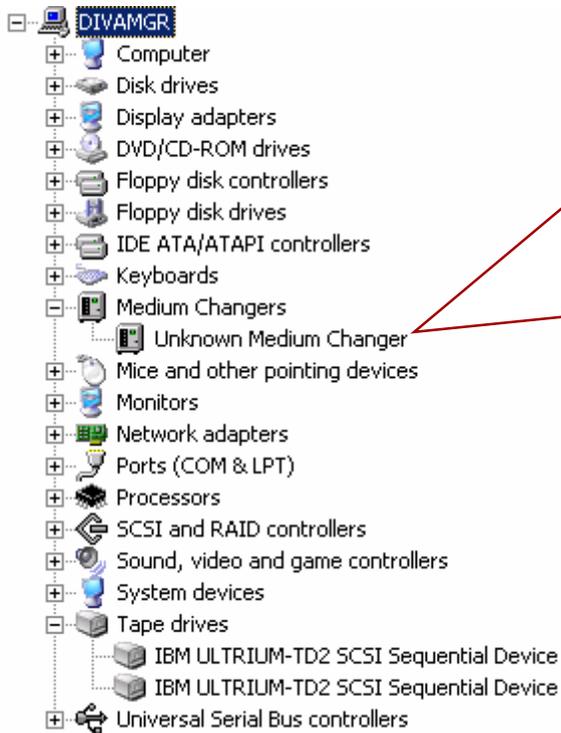
Figure 76: Determining the SCSI Library Connection Using the Scandrive Utility



The **Type** section refers to that peripheral's class (*HDD, CDROM, etc.*). A tape library will be reported as a **Medium Changer Peripheral**, and the **Identifier** for each corresponding device reported should match the model number of the library itself (e.g. *SL500*). The full SCSI path reported for each library can then be entered to the `RM_SCSI_DEVICE_LSM(x)` settings in the `RobotManager.conf` file.

If you cannot locate a specific library in the Scandrive Utility, but that library is visible in your HBA, then the library has likely been disabled in the Windows Device Manager (denoted by an **'X'** over the device icon). You must re-enable the device for it to appear in the Scandrive Utility.

Figure 77: Windows Device Manager



Caution! For most SCSI interface libraries the DIVArchive Robot Manager communicates to the library directly over the SCSI hardware layer and does not require the traditional Windows driver interface.

The exceptions however are for IBM and HP libraries.

For all other libraries, it is essential that no Windows library driver be loaded for the library interface, otherwise the DIVArchive Robot Manager will not be able to communicate with the library. In this case, if your library does not appear in Windows Device Manager as an **Unknown Medium Changer**, the Robot Manager will be unable to communicate correctly with the robotics.

4.3 Sony ODA Drives

Sony ODA ODS-D55U and ODS-D77F Drives are supported by DIVArchive starting with version 7.2. These are Blu-ray Optical Drives and the media is WORM (*Write-Once-Read-Many*) media using a UDF format, and only AXF formatted objects may be written to the discs. The drives are controlled by the Robot Manager and the media is viewed as a Tape Cartridge.

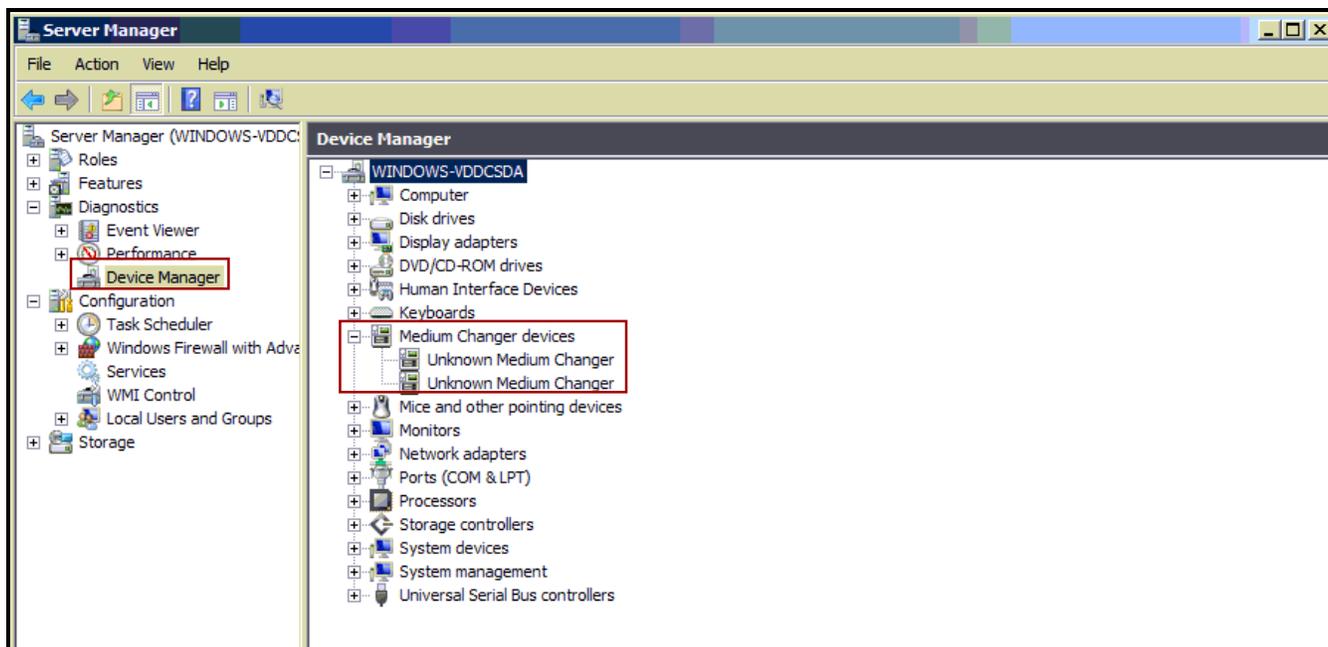
In the Windows Device Manager these drives will be shown as **Unknown Medium Changer** under the **Medium Changer devices** section since there are no device drivers for them. The drive itself will also appear as an **Optical SCSI Device** with the make and model number under the **Disk Drives** section.

There are six different types of disc media available for use with the Sony Optical Drives:

- SONY-ODC300R
- SONY-ODC300RE
- SONY-ODC600R
- SONY-ODC600RE
- SONY-ODC1200RE
- SONY-ODC1500R

The disc types are identified in the `scsi_tape_types.ini` file (*described below*).

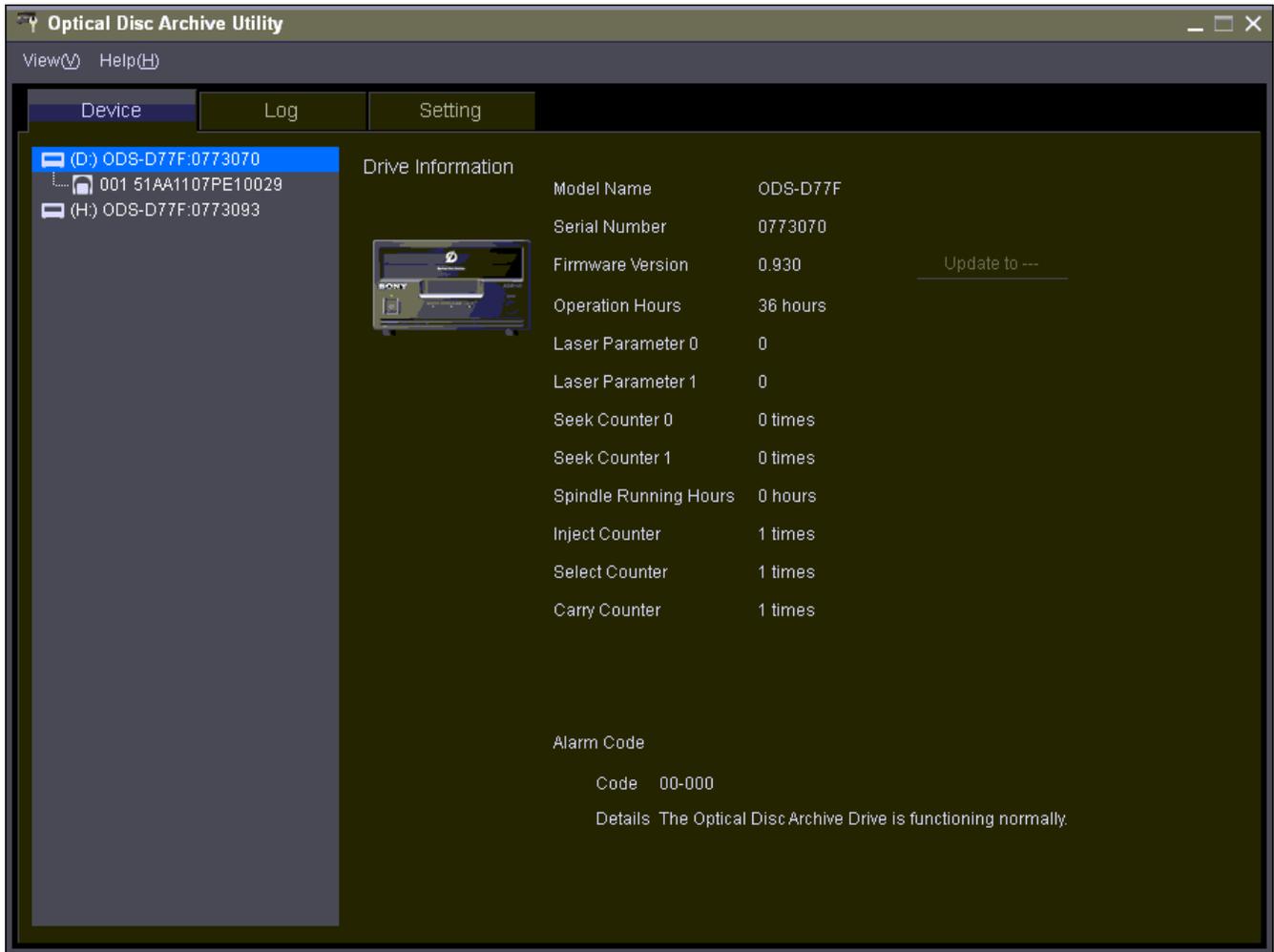
Figure 78: Windows Device Manager Showing Sony ODA Drives



The drive specifics may be viewed using the Optical Disc Archive Utility as shown in the figure below. The utility allows viewing of the device logs as well as viewing and changing of the drive settings.

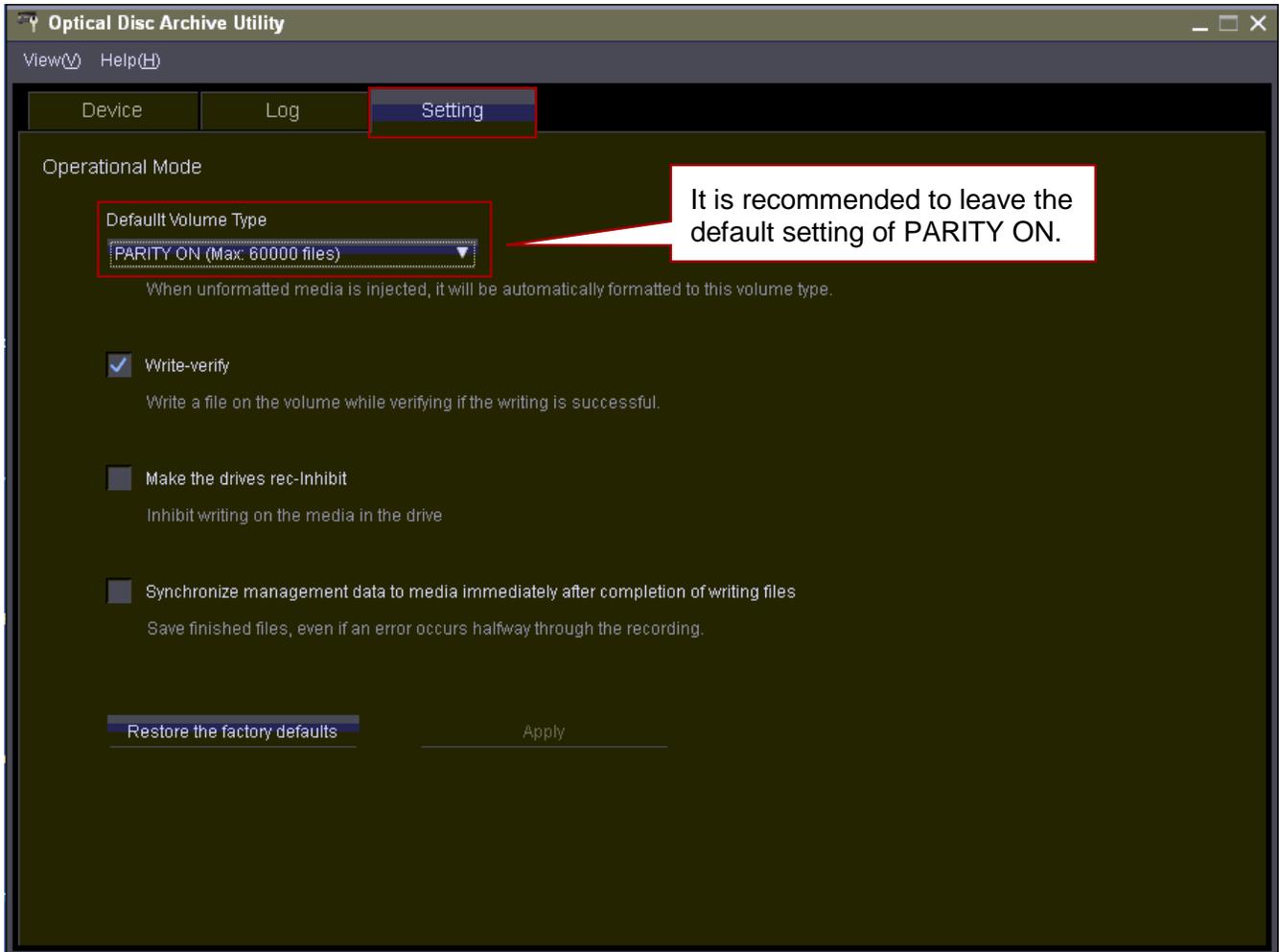
Note: The Drive Settings must be configured BEFORE configuring DIVArchive. The recommended parity setting is PARITY ON.

Figure 79: Optical Disc Archive Utility



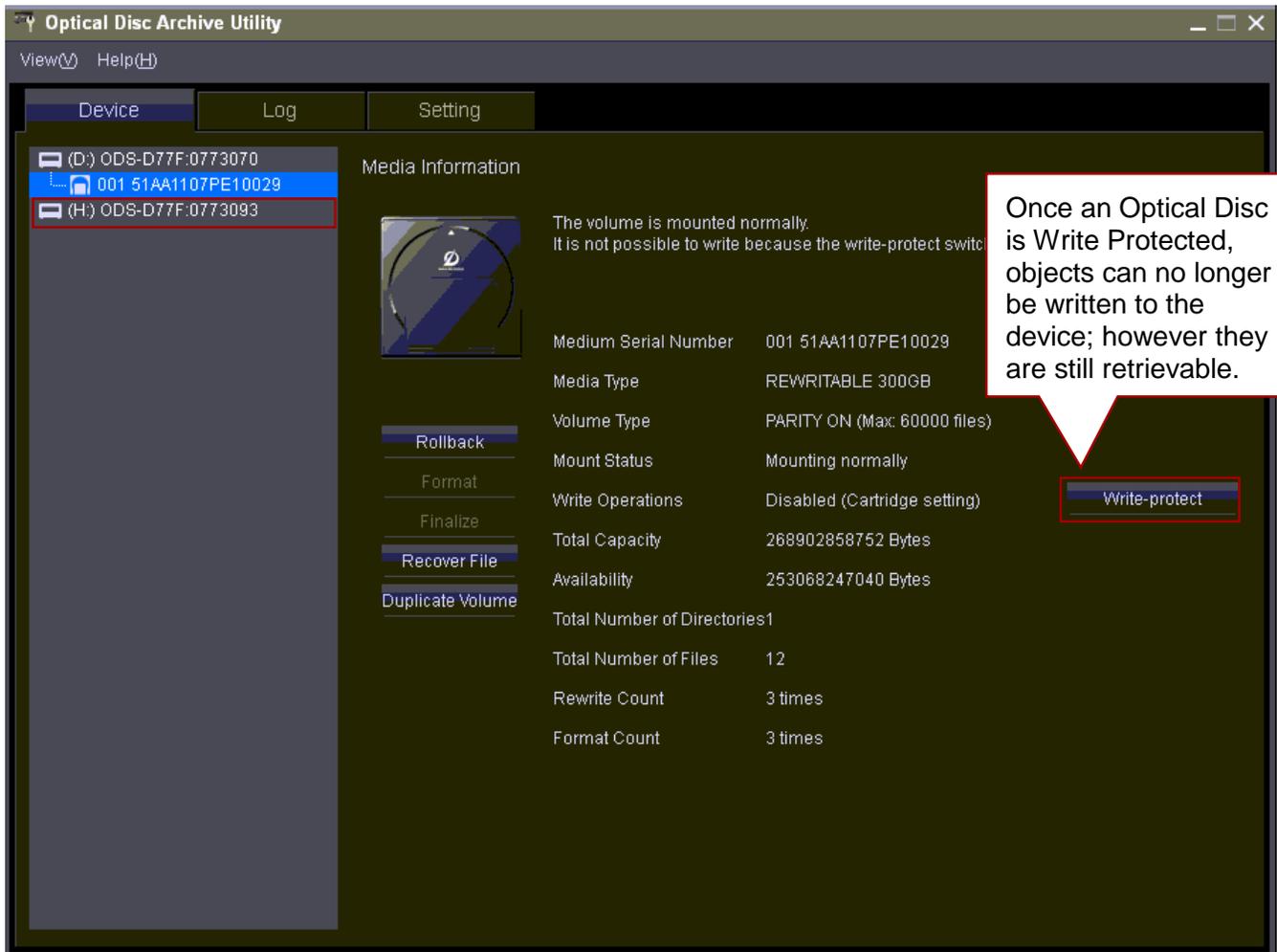
To change the drive settings, click on the **Setting Tab** in the **Optical Disc Archive Utility** as shown below. It is recommended to leave the **Default Volume Type** set to **PARITY ON**, and to use the default settings for the remaining items.

Figure 80: Optical Disc Archive Utility Drive Settings Tab



To view information about the Media in a Drive, click on the **Media Listing** under the **Drive Listing**. The displayed information will be similar to the screen shown in the figure below.

Figure 81: Optical Media Information Screen



4.3.1 Configuration File Changes

There are several changes that need to be made to the configuration files.

- In the `robotmanager.conf` configuration file, under the **SCSI module specific options**, the serial number must be identified:

```
# SCSI host devicename or serial number
# Ex: ScsiP:B:T:L (for Windows)
#       -P Scsi port
#       -B Bus number
#       -T Target ID of the changer : Integer [0..255]
#       -L Logical Unit Number of the changer : Integer [0..7]
# If this parameter contains the serial number of the library, the
# SCSI RobotManager will scan all the SCSI devices in order to
# find the right device name automatically.
RM_SCSI_DEVICE_LSM0=00001003
# RM_SCSI_DEVICE_LSM1 must be used only to control the second
# frame of a dual L700e with pass through port
#RM_SCSI_DEVICE_LSM1=
```

- In the `scsi_drive_types.ini` file, the D55U and D77F drive types must be uncommented (*remove the #*):

```
#-----
# If the SCSI Robot Manager is connected to a SONY ODA library
# UNCOMMENT ALL LINES IN THE FOLLOWING PART
#-----
#TypeID TransportDomain TransportType TypeName CompatibleTapeTypes
#-----
---
#600 0x00 0x00      SONY-ODS-D55U      600 601 602 603 604 605
#601 0x00 0x00      SONY-ODS-D77F      600 601 602 603 604 605
```

Remove the # mark to uncomment these drive types for whichever drives are in the system.

The `TransportDomain` and `TransportType` are obtained automatically and not used in the configuration – therefore leave these set to 0x00 as shown.

- In the `scsi_tape_types.ini` file, uncomment all of the disc types listed (*remove the #*). The **R** or **RE** at the end of the disc number indicates whether the disc is **Write Once (R)** or **Rewritable (RE)**. This indicator is used because the barcode does not contain the video type as in normal tape barcodes.

```
#-----
```

```

# If the SCSI Robot Manager is connected to SONY ODA library,
# UNCOMMENT ALL LINES IN THE FOLLOWING PART
#-----
#TypeID TransportDomain TransportType TypeName CompatibleDriveTypes
#-----
#600 0x00 0x00      SONY-ODC300R      600 601
#601 0x00 0x00      SONY-ODC300RE     600 601
#602 0x00 0x00      SONY-ODC600R      600 601
#603 0x00 0x00      SONY-ODC600RE     600 601
#604 0x00 0x00      SONY-ODC1200RE    600 601
#605 0x00 0x00      SONY-ODC1500R     600 601

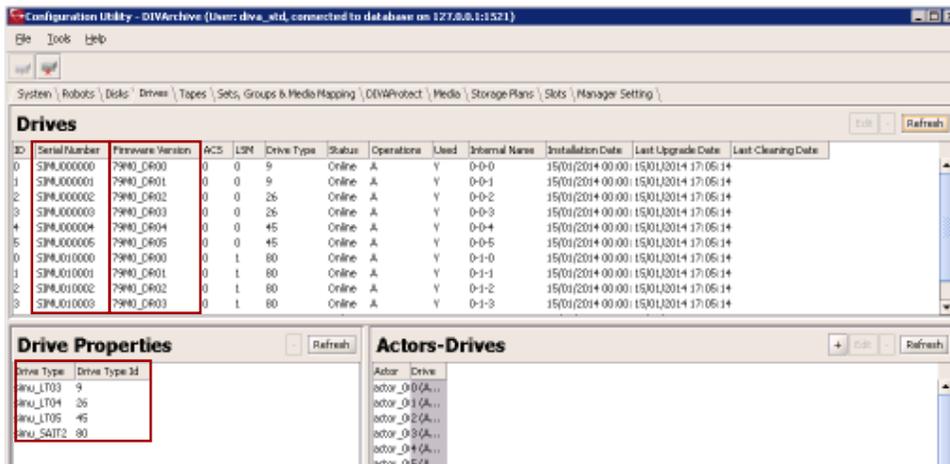
```

4.3.2 Configuration Utility Settings

The following settings should be configured in the DIVArchive Configuration Utility.

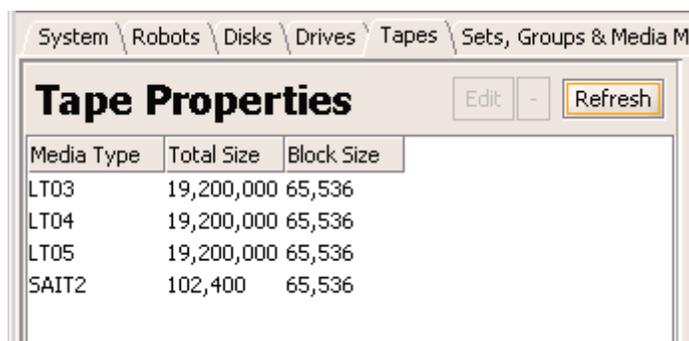
- **Drives Tab**
 - Set the **Drive Properties** to 64k.
 - The Serial Number comes from the Robot Manager.
 - The Firmware Version comes directly from the Drive.

Figure 82: Configuration Utility Drives Tab



- **Tapes Tab**
 - The Tape Properties Pane shows all of the (*uncommented*) Tape Types from the `scsi_tape_types.ini` file.

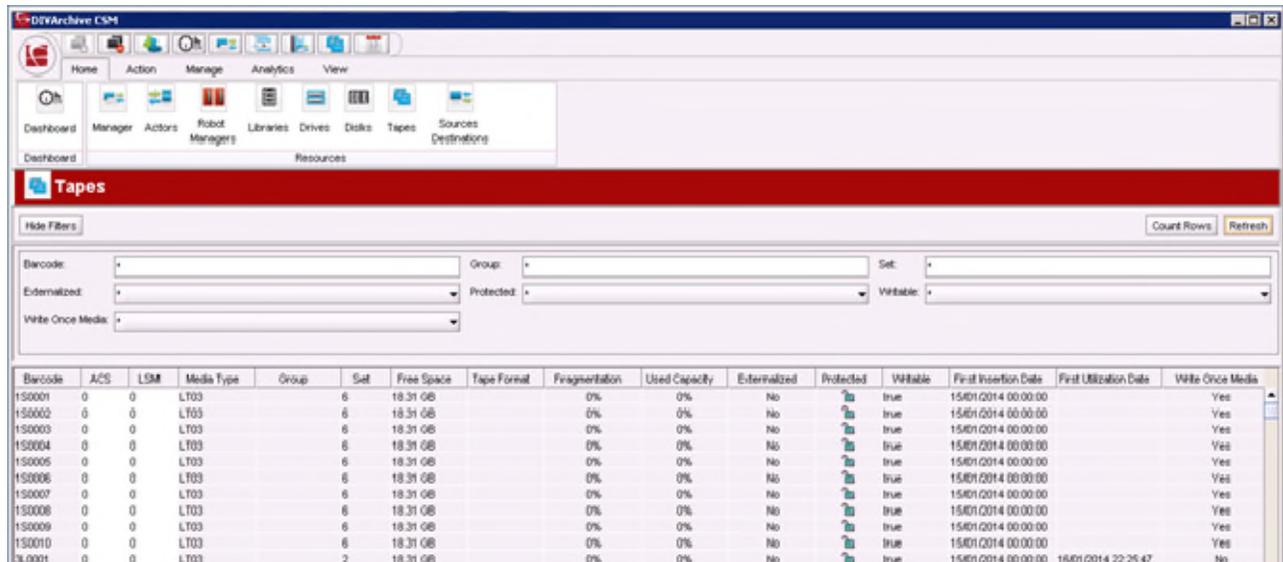
Figure 83: Configuration Utility Tapes Tab – Tape Properties Pane



4.3.3 Control GUI Settings and Information

The Optical Drives and Discs are seen in the DIVArchive Control GUI as Tape Drives and Tapes respectively.

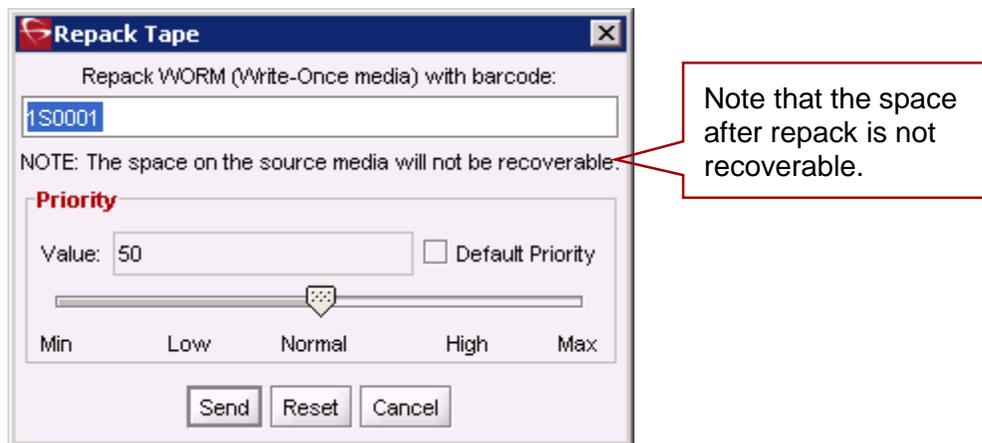
Figure 84: DIVArchive Control GUI Tapes Window



Barcode	ACS	LSM	Media Type	Group	Set	Free Space	Tape Format	Fragmentation	Used Capacity	Externalized	Protected	Writable	First Insertion Date	First Utilization Date	Write Once Media
150001	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150002	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150003	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150004	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150005	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150006	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150007	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150008	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150009	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150010	0	0	LTO3	6	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00		Yes
150001	0	0	LTO3	2	18.31 GB			0%	0%	No		true	15/01/2014 00:00:00	15/01/2014 22:25:47	No

Repack of the discs and Deletion of object is available; however, the space is not recoverable. When trying to repack the disc, the typical Repack Window will be displayed but there will be a warning that the space is non-recoverable. Due to this limitation of the discs, auto-repack has been disabled for these drives and discs.

Figure 85: Repack Window with Warning

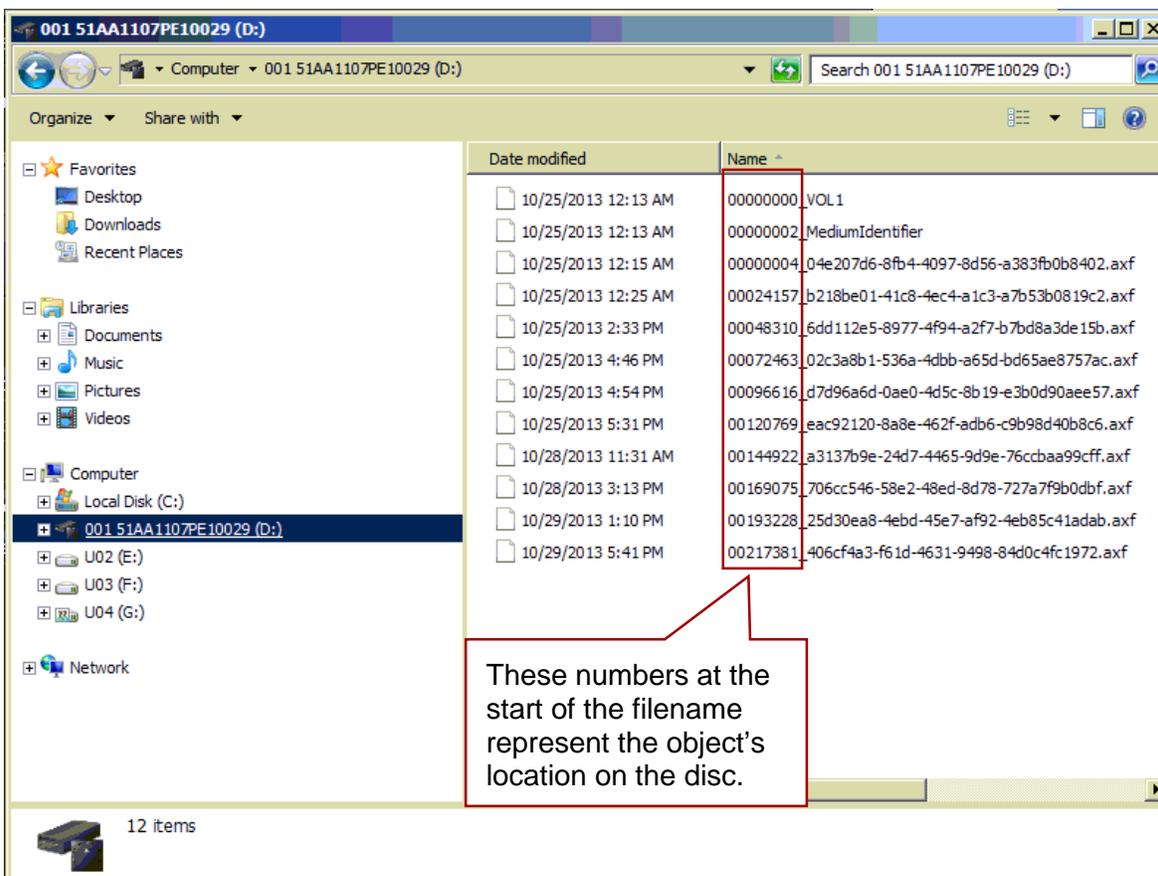


4.3.4 Additional Information

Additional information related to the use of the Optical Drives and Discs:

- Within the DIVArchive Control GUI the Optical Discs are displayed under the **Drives Tab**.
- Because Write-Once Media must be finalized, zero remaining space will be reported to the Manager.
- Objects are spanned when there is 100mb is remaining so that there is space left and the disc may be finalized. Once an object is spanned, the disc is considered full and is automatically finalized.
- The Actor will auto-finalize the discs when there is 500mb of space remaining (*unless an object was spanned*); however it can be achieved manually through the Optical Disc Archive Utility.
- If a drive is manually mounted and viewed in the Windows Explorer, the display will be similar to the figure below.
 - The numeric value at the beginning of each object's filename identifies the object's location on the tape.

Figure 86: Viewing a Mounted Drive in Windows Explorer



4.4 Configuring Direct Attached SCSI Libraries

A Direct Attached Library is directly connected to the DIVArchive Robot Manager host either through a native SCSI interface and SCSI Host Bus Adaptor (HBA), or via a SCSI over Fiber Channel connection and Fiber Channel HBA (either directly or via a SAN).

In either case, the DIVArchive Robot Manager uses its own DIVArchive provided driver (*SCSI_Robot.dll* or *libSCSI_Robot.so* depending on the host's Operating System) to directly interface with the library without the need for intermediate library management software. For this type of SCSI attached library, the following entries need to be un-commented and/or configured in the `RobotManager.conf` file.

Drive Models and Tape Types for libraries are covered in other configuration files (see [Section 7](#)).

4.4.1 Common Settings for SCSI Based Libraries

Table 10: Common Settings for SCSI Based Libraries

RobotManager Common Options	Operating System	Action
<code>RM_MODULE=SCSI_Robot.dll</code>	Windows	Uncomment this line only.

Table 11: SCSI Device Parameters

Parameter	Parameter Type	Default
<code>SERVICE_NAME</code>	Name	Uncommented
Display Name of the Robot Manager Windows Service. This variable must be set if multiple Robot Managers are to be installed on the same server. If this variable is used, the Service Name will be <code>DivarBt-<SERVICE_NAME></code> . If this variable is not set, the Service Name will revert to <code>DivarBt</code> .		
<code>RM_PORT</code>	TCP Port Number	8500
The TCP port that the DIVArchive Robot Manager will listen to for incoming requests. This value must be unique if there are multiple DIVArchive Robot Managers running on the one host. Typically, TCP port 8500 and up.		
<code>RM_ACS</code>	Number	0
Automated Cartridge System (ACS) controlled by the DIVArchive Robot Manager Module. This value will appear in the Robot Manager/ACS Association List in the		

Parameter	Parameter Type	Default
Configuration Utility for this DIVArchive Robot Manager after Database Synchronization.		

Table 12: SCSI Module Parameters

SCSI Module Parameters	Operating System	Parameter
RM_SCSI_DEVICE_LSM0	Windows	ScsiP:B:T:L
This specifies the SCSI target of the library as it identified by the host operating system. Refer to Section 4.2.2 to determine these values.		
RM_SCSI_DEVICE_LSM1	Windows	ScsiP:B:T:L
This setting is specific to a SUN (<i>StorageTek</i>) dual L1400M library with a Pass Through Port (<i>PTP</i>), and specifies the SCSI target of the 2 nd frame (<i>LSM</i>). Although this type of library configuration could be addressed from only the RM_SCSI_DEVICE_LSM0 connection, DIVArchive manages this type of library more effectively when both frames are specified. DIVArchive also manages the PTP in this case.		

4.4.2 Additional Settings for Breecehill Libraries

Table 13: Additional Settings for Breecehill Libraries

SCSI Module Parameters	Parameter Type	Default
RM_SCSI_SKIP_DIVAWORK_AUDIT	0 (<i>disabled</i>) 1 (<i>enabled</i>).	0
<p>When the Robot Manager is launched, this option will enable a Read Element Status command to be issued to the library to audit and synchronize the content of the library's internal memory. This operation takes approximately one minute. This feature can be disabled by enabling this parameter (<i>set it to 1</i>).</p>		
RM_SCSI_ENABLE_MEDIA_TYPE_DETECTION	0 (<i>disabled</i>) 1 (<i>enabled</i>)	0
<p>This option specifies whether the Tape Media Type may be embedded in the barcode labels returned by the library. A barcode label is usually a 6 digit string, however if this feature is enabled, these libraries return an 8 digit string. The two additional characters are not part of the actual label but are used to define the media type. If this value is enabled, the DIVArchive Robot Manager will determine the Tape Media Type by using the label and remove the two additional characters from the label.</p>		
RM_SCSI_MEDIA_TYPE_LEFT_DETECTION	0 (<i>disabled</i>) 1 (<i>enabled</i>)	1
<p>This option indicates to the DIVArchive Robot Manager the location of the extra two characters if the RM_SCSI_ENABLE_MEDIA_TYPE_DETECTION option has been enabled. Depending on the specific library, the additional 2 characters may be on the right or on the left of the label. If the media type information is on the left, set this value to 1 (<i>enabled</i>), otherwise set it to 0 (<i>disabled</i>).</p>		

4.4.3 Additional Settings for Qualstar and ADIC (Return Media Id Enabled) Libraries

Table 14: Additional Settings for Qualstar and ADIC Libraries

SCSI Module Parameters	Parameter Type	Default
RM_SCSI_ENABLE_MEDIA_TYPE_DETECTION	0 (<i>disabled</i>) 1 (<i>enabled</i>)	0
<p>This option specifies whether the Tape Media Type may be embedded in the barcode labels returned by the library. A barcode label is usually a 6 digit string, however if this feature is enabled, these libraries return 8 digit strings. The two additional characters are not part of the actual label but are used to define the media type. If this value is enabled, the DIVArchive Robot Manager will determine the Tape Media Type by using the label and remove the two additional characters from the label.</p>		
RM_SCSI_MEDIA_TYPE_LEFT_DETECTION	0 (<i>disabled</i>) 1 (<i>enabled</i>)	1
<p>This option indicates to the DIVArchive Robot Manager the location of the extra two characters if the RM_SCSI_ENABLE_MEDIA_TYPE_DETECTION option has been enabled. Depending upon the specific library, the additional 2 characters may be on the right or on the left of the label. If the media type information is on the left, set this value to 1 (<i>enabled</i>), otherwise, set it to 0 (<i>disabled</i>).</p>		

4.4.4 Additional Settings for DVD Jukeboxes (ASACA or Plasmon)

4.4.4.1 DVD Jukeboxes Limitations

The primary difference between a DVD Jukebox and a regular Tape Library is that the former does not implement the `READ VOLUME ELEMENT ADDRESS` SCSI command. As a result, the media in the Jukebox is identified by their element address and not their volume tag or label. Conventional tape libraries allow the element location of a barcode ID to be provided by the library, but DVD Jukeboxes do not provide this functionality. Due to this limitation of the DVD Jukeboxes, the DIVArchive Robot Manager for this library instance needs to maintain its own local media database of element addresses.

Note: Intermediate Windows drivers (if loaded) for the Jukebox may cause an issue with the control of the Jukebox by the DIVArchive Robot Manager. It is recommended these driver not be loaded, or if they are loaded, disable them using the Windows Device Manager.

4.4.4.2 RobotManager.conf Options

Table 15: RobotManager.conf Options

SCSI Module Parameters	Parameter Type	Default
<code>RM_SCSI_AUDIT_JUKEBOX</code>	0 (<i>disabled</i>) 1 (<i>enabled</i>)	0
For ASACA or PLASMON DVD Jukeboxes, set this value to 1 to audit the Jukebox and populate the JukeboxDB Database. This should only be disabled at the completion of the audit.		
<code>RM_SCSI_AUDIT_DRIVE</code>	<code>ScsiP:B:T:L</code>	<code>Scsi2:0:5:0</code>
This parameter contains the SCSI path of the audit drive. The syntax is <code>ScsiP:B:T:L</code> where: P = Port Number B = Bus number T = Target ID L = Logical Unit Number.		
<code>RM_SCSI_AUDIT_DRIVE_ID</code>	0 (<i>disabled</i>) 1 (<i>enabled</i>)	0
<code>RM_SCSI_AUDIT_DRIVE_ID</code> is the Drive Number that will be reported to the Configuration Utility.		

SCSI Module Parameters	Parameter Type	Default
RM_SCSI_MEDIA_TYPE	Number	0
This parameter sets the default Media Type ID (<i>refer to SCSI_Tape_Type.ini for the appropriate values</i>).		
RM_SCSI_UPDATE_LABELS	0 (<i>disabled</i>) 1 (<i>enabled</i>)	1
When the media is audited, the parameter defines whether DIVArchive will incorporate the existing DVD label into its database, or create a new label. If this option is enabled, a new label will be written and this will invalidate any existing data on the media.		

4.4.4.3 Creating or Updating the JukeboxDB Database

The local media database is a binary file called `JukeboxDB`, which once created, is stored in the same directory as the DIVArchive Robot Manager executable files. This must first be initially populated by enabling the `RM_SCSI_AUDIT_JUKEBOX` parameter in the Robot Manager Configuration file.

To successfully complete the audit, the full SCSI path of one of the DVD drives must be configured in the `RM_SCSI_AUDIT_DRIVE_ID` section in the Robot Manager Configuration file (*refer to Section 4.2.2, which covers the Windows SCSI Tree for determining the full SCSI path of the drive as it appears to Windows*). This drive will also be used to audit new media as it is entered into the DVD Jukebox.

Additionally, you must also enter the Media Type settings for the Jukebox; these can be extracted from the `scsi_TapeTypes` file and entered into the `RM_SCSI_MEDIA_TYPE` parameter.

4.5 Configuring ACSLS Attached Libraries

Although DIVArchive can directly interface to most SUN (*StorageTek*) libraries using the `Robot_scsi` driver, some library configurations may require the use of SUN's ACSLS library management software for the Robot Manager to control the library.

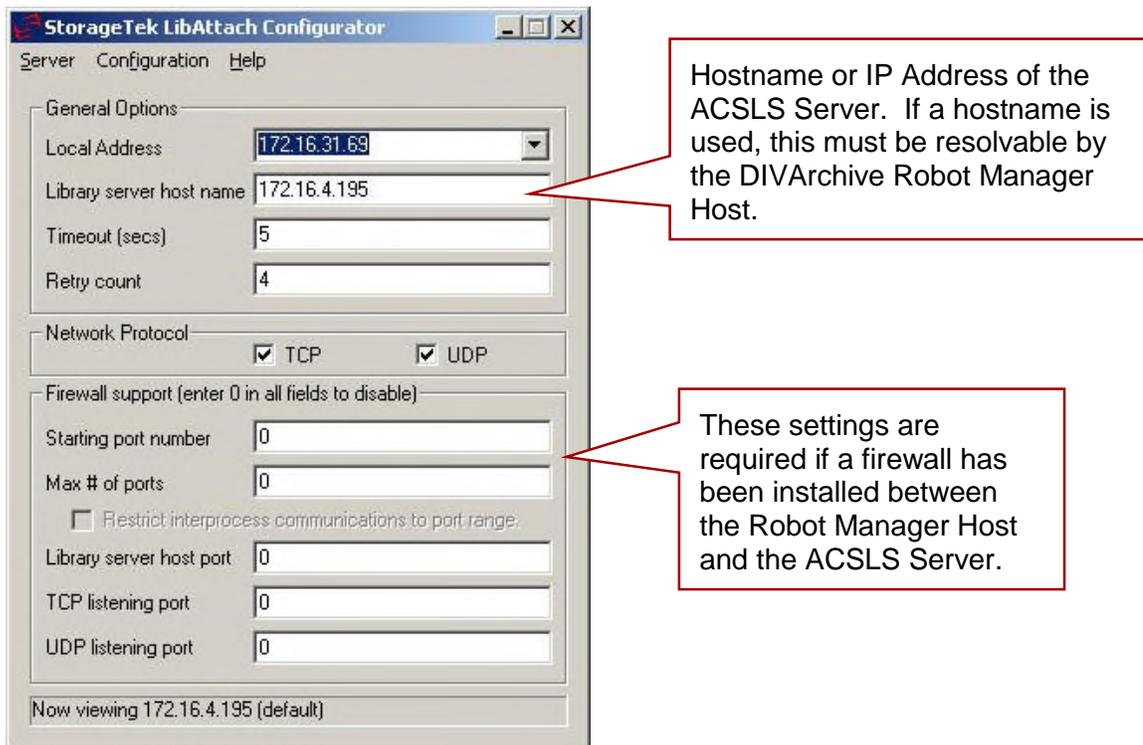
ACSLs is an acronym for Automated Cartridge System Library Software, and can only be installed on Solaris platforms. The Solaris host and ACSLS are sold and supported by SUN Microsystems (*formerly StorageTek*). Refer to the Oracle ACSLS documentation for specific information regarding this product.

Note: DIVArchive installations under the Solaris Operating System are no longer supported by Oracle starting with DIVArchive version 6.5.0 and the Linux Operating System starting with DIVArchive version 7.1.0.

4.5.1 LibAttach Configuration

Connectivity to the ACSLS host is provided by an intermediate Windows driver called `LibAttach`. This is normally installed on the same host that is running the DIVArchive Robot Manager and runs as a Windows Service. The DIVArchive Robot Manager communicates to the ACSLS host via the `LibAttach` driver.

Figure 87: LibAttach Configuration



4.5.2 Testing LibAttach Connectivity to ACSLS

In the `LibAttach` installation directory, a small utility named `query_server.exe` can be used to verify connectivity from the Robot Manager Host to the ACSLS Server and software.

When this utility is launched, a command prompt window will open. If the connection is successful, statistics from the library will be returned.

4.5.3 Firewall Support Notes

If a network firewall is between your Robot Manager Host and ACSLS Server you will need to have a TCP or UDP port opened to allow communication between the two. If this is true, enter the port numbers into the **Firewall Support** settings in the `LibAttach` Configurator.

Note: The early implementation of firewall support for `LibAttach` did not work correctly with the DIVArchive Robot Manager even though the `query_server` utility returned a successful connection. SUN Microsystems issued a patch to resolve this issue. Please ensure that you have the latest release of `LibAttach` that incorporates this patch. Contact Oracle Technical Support for further information.

4.5.4 RobotManager.conf Common Options

Table 16: RobotManager.conf Common Options

RobotManager Common Options	Parameter Type	Default
RM_MODULE=ACSLs_Robot.dll	Uncomment only this line.	
RM_PORT	TCP Port Number	8500
The TCP Port that the DIVArchive Robot Manager will listen on for incoming requests. This value must be unique if there are multiple DIVArchive Robot Managers running on the one host. Typically, TCP Port 8500 and up.		
RM_ACS	Number	Ignored
For ACSLS configurations, this value is ignored as the ACS number is supplied from ACSLS.		
SERVICE_NAME	Name	Uncommented
Display Name of the Robot Manager Windows Service. This variable must be set if multiple Robot Managers are to be installed on the same server. If this variable is used, the Service Name will be <code>Divarbt-<SERVICE_NAME></code> . If this variable is not set, the Service Name will revert to <code>Divarbt</code> .		

Table 17: ACSLS Parameters

ACSLs Parameters	Parameter Type	Default
RM_ACSLS_SERVER	IP Address or Hostname	Ignored and can be left blank.
RM_ACSLS_SSI_SOCKET	TCP Port Number	
ACSLs SSI socket is the UNIX domain socket used by SSI. If this value is left undefined TCP port 50004 is used by default.		

ACSLS Parameters	Parameter Type	Default
RM_ACSLS_TIMEOUT	Time in milliseconds	0
<p>This sets the timeout period for queries to ACSLS (<i>via LibAttach</i>). If this value is left at 0, the timeout period used by the Robot Manager is 10 minutes. If this timeout needs to be altered, replace 0 with your own value (<i>in milliseconds</i>).</p>		
RM_ACSLS_IE_TIMEOUT	Time in milliseconds	0
<p>When an insert or eject tape command is issued the operator needs to open the CAP and insert or eject tapes within this timeout. If this value is left at 0, the timeout period used by the Robot Manager is 10 minutes. If this timeout needs to be altered, replace 0 with your own value (<i>in milliseconds</i>).</p>		
RM_ACSLS_MAX_DISMOUNT_RETRIES	Number of retries	5
<p>Maximum number of retries when the dismounted drive is still in use. The default value is 5. The initial delay is 5 seconds and then doubled after each retry.</p>		
RM_ACSLS_DISMOUNT_FORCE	0 (<i>disabled</i>) 1 (<i>enabled</i>)	0
<p>Under normal circumstances a tape must first be unloaded (<i>via an Actor</i>) prior to issuing a dismount command to the library. A forced dismount will instruct the library to issue the unload command to the drive directly. This option is not recommended as this may interfere with operations on the Actor(s).</p>		

4.6 Configuring Sony PetaServe Libraries

Control of Sony PetaServe libraries from the DIVArchive Robot Manager is directed through the Sony PSC controller over an Ethernet connection.

The PSC controller parameters for the Robot Manager Configuration file should match those on the PetaSite Controller.

4.6.1 RobotManager.conf Common Options

Table 18: RobotManager.conf Common Options

RobotManager Common Options	Parameter Type	Default
<code>RM_MODULE=SONY_Robot.dll</code>	Uncomment only this line.	
<code>RM_PORT</code>	TCP Port Number	8500
<p>The TCP port that the DIVArchive Robot Manager will listen on for incoming requests. This value must be unique if there are multiple DIVArchive Robot Managers running on the one host. Typically, TCP port 8500 and up.</p>		
<code>RM_ACS</code>	Number	0
<p>Automated Cartridge System (ACS) controlled by the Robot Manager Module.</p>		

Table 19: SONY PetaSite Parameters

SONY PetaSite Parameters	Parameter Type	Default
<code>RM_SONY_ENABLE_MEDIA_TYPE_TRIMMING</code>	Number	1
<p>Some tape labels contain an additional 2 or 3 characters identifying the type of media. Examples: 004452L2 is an LTO-2 tape. S1000052 is a SAIT1 tape. If this parameter is set to 1, the Sony Robot will detect the tape using the label and filter out the 2 or 3 additional characters from the label. WARNING: This parameter should not be modified during production. If it is changed during production the database may need to be patched.</p>		
<code>RM_SONY_MEDIA_TYPE_TRIMMING_LEFT</code>	Number	0
<p>Depending upon the label, the 2 characters may be on the right or on the left of the label. If the Media Type information is on the left, set this parameter to 1; otherwise set it to 0. WARNING: This parameter should not be modified during production. If it is changed during production the database may need to be patched.</p>		

SONY PetaSite Parameters	Parameter Type	Default
RM_SONY_PSCSERVERNAME	IP Address or Hostname	
<p>This specifies the hostname or IP Address of the Sony PetaSite controller (<i>PSC</i>). If a hostname is specified, this must be defined in the <code>hosts</code> file for the operating system.</p>		
RM_SONY_PSCUSERID	Number	1
<p>This specifies the <code>user ID</code> that the Robot Manager will specify when it connects to the Sony PetaSite Controller.</p>		
RM_SONY_PSCTIMEOUT	Time in milliseconds	900000
<p>Command time out to the PSC. The default is 900000 milliseconds or 15 minutes. This is only used for mount operations.</p>		
RM_SONY_PSCDISMOUNTRETRIES	Number of retries	5
<p>Maximum number of retries when the dismounted drive is still in use. If the setting is 5, the initial delay is 5 seconds and then doubled after each retry.</p>		

4.7 Configuring ADIC Libraries with SDLC

This interface is only available on Windows platforms. Refer to APPENDIX C – ADIC SDLC Installation Guide for setting up the SDLC Server and Client Components for the DIVArchive Robot Manager interface.

4.7.1 RobotManager.conf Common Settings

Table 20: RobotManager.conf Common Settings

RobotManager Common Options	Parameter Type	Default
<code>RM_MODULE=ADIC_Robot.dll</code>	Uncomment only this line.	
<code>RM_PORT</code>	TCP Port Number	8500
The TCP port that the DIVArchive Robot Manager will listen on for incoming requests. This value must be unique if there are multiple DIVArchive Robot Managers running on the one host. Typically, TCP port 8500 and up.		
<code>RM_ACS</code>	Number	0
Automated Cartridge System (ACS) controlled by the Robot Manager Module.		

Table 21: ADIC Parameters

ADIC Parameters	Parameter Type	Default
<code>RM_ADIC_DAS_CLIENT</code>	Hostname	
Hostname of the machine running the ADIC DAS client.		
<code>RM_ADIC_EJECT_AREA_NAME</code>	Name	E01
Symbolic name of the Cartridge Access Port.		
<code>RM_ADIC_TIME_INSERT</code>	Time in milliseconds	5000
Number of milliseconds to wait to put away the tape after closing the CAP. The default value is 5000ms.		

ADIC Parameters	Parameter Type	Default
<code>RM_ADIC_MAX_DISMOUNT_RETRIES</code>	Number of retries	5
Maximum number of retries when the dismounted drive is still in use. If the setting is 5, then initial delay is 5 seconds and then doubled after each retry.		

4.8 Configuring Simulated Libraries (for DIVArchive Simulators)

Simulated Robots are only available on Windows platforms. These settings are shown here for reference only. Refer to the *Oracle DIVArchive Simulator Operations Guide* (available to OPN partners only) for more information on installing and configuring a DIVArchive Simulator platform.

4.8.1 RobotManager.conf Common Settings

Table 22: RobotManager.conf Common Settings

RobotManager Common Options	Parameter Type	Default
RM_MODULE=SIMULATOR_Robot.dll	Uncomment only this line.	
RM_PORT	TCP Port Number	8500
The TCP port that the DIVArchive Robot Manager will listen on for incoming requests. This value must be unique if there are multiple DIVArchive Robot Managers running on the one host. Typically, TCP port 8500 and up.		
RM_ACS	Number	0
Automated Cartridge System (ACS) controlled by the Robot Manager Module.		

Table 23: Simulator Parameters

Simulator Parameters	Parameter Type	Default
RM_SIMU_BASEDIR	Directory path	
This is the base directory path to where the DIVArchive simulation files are located (<i>typically C:\Diva\Simulation</i>).		
RM_SIMU_OPERATION_SHORT_DELAY	Time in milliseconds	0
This setting is used to simulate physical delays in mount/dismount/enter/eject operations. The recommended setting is 10000ms.		

Simulator Parameters	Parameter Type	Default
RM_SIMU_OPERATION_LONG_DELAY	Time milliseconds	in 0
This setting can be used to simulate an operation that takes more time than expected for execution. The recommended setting is 120,000ms.		
RM_SIMU_OPERATION_LONG_DELAY_FREQUENCY	Number	0
This setting specifies how often a long delay should occur. The recommended setting is 50.		
RM_SIMU_LIST_SHORT_DELAY	Time milliseconds	in 0
This setting introduces a simulated physical delay in list operations. The recommended setting is 500.		
RM_SIMU_LIST_LONG_DELAY	Time milliseconds	in 0
This setting can be used to simulate an operation that takes more time than expected for execution in list operations. The recommended setting is 60000ms.		
RM_SIMU_LIST_LONG_DELAY_FREQUENCY	Number	0
This setting specifies how often a long delay should occur in list operations. The recommended setting is 100.		

4.9 Robot Manager Command Options

Control and management functions of the DIVArchive Robot Manager on Windows platforms are performed by utilizing `Robotmanager.exe` from a Windows command prompt. The executable is located in the `DIVA_HOME\Program\Robotmanager\bin` folder.

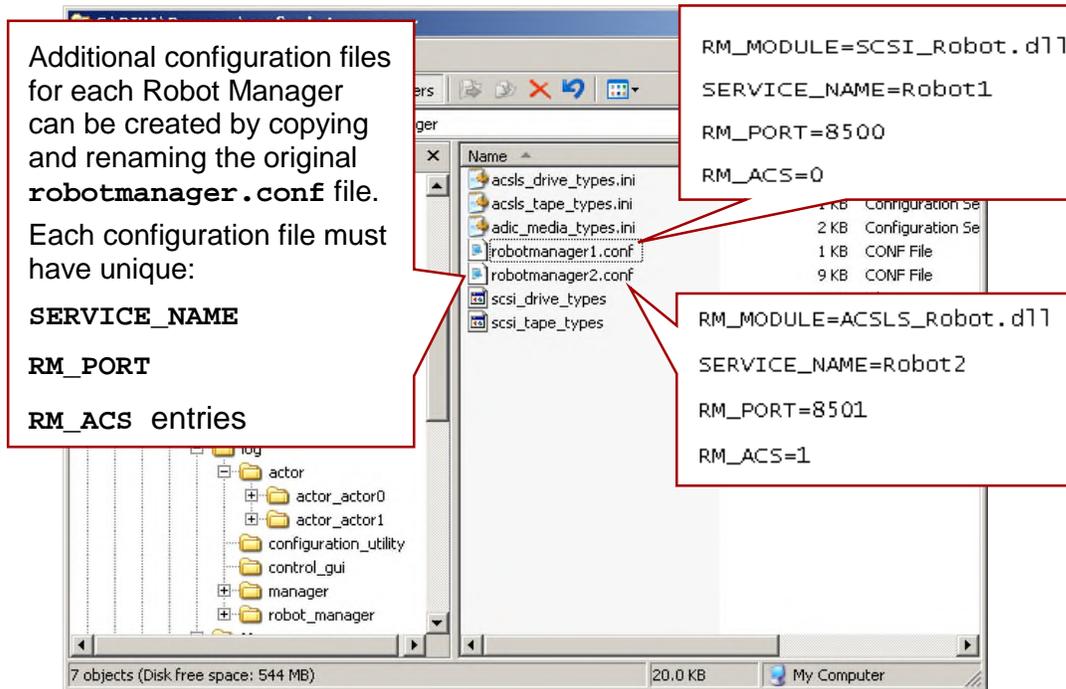
4.9.1 Installing/Removing the RobotManager Service(s)

The following command line options are used to install (*or uninstall*) the DIVArchive Robot Manager as a Windows System Service from a Windows command prompt.

- | | |
|------------------------------|---|
| <code>robotmanager -i</code> | Installs the Robot Manager Service as set by the <code>service_name</code> parameter defined in <code>robotmanager.conf</code> . If this parameter is undefined, the service is installed as <code>DIVArchive Robot Manager - hostname</code> . |
| <code>robotmanager -u</code> | Removes the Robot Manager Service set by the <code>service_name</code> parameter in <code>robotmanager.conf</code> . If this parameter is undefined the service to be removed is <code>DIVArchive Robot Manager - hostname</code> . |

The above Robot Manager command options default to the `robotmanager.conf` file located in the `DIVA_HOME\Program\conf\robot_manager` folder to define the Service Name (*if any*). If you are installing multiple Robot Managers on a single host, additional Robot Manager Configuration Files must be created and specified to the service during installation to create unique instances for each Robot Manager.

Figure 88: Installing/Removing the Robot Manager Service(s)



When installing additional Robot Manager Services on the same host, the path to each Robot Manager's Configuration File needs to be specified for each instance. This is done by adding the **-conf** (or **-f**) command switches when installing the service:

```
robotmanager -i -conf ..\..\conf\robot_manager\robotmanager2.conf
```

Installs the Robot Manager Service as defined by the **service_name** parameter in **robotmanager2.conf**.

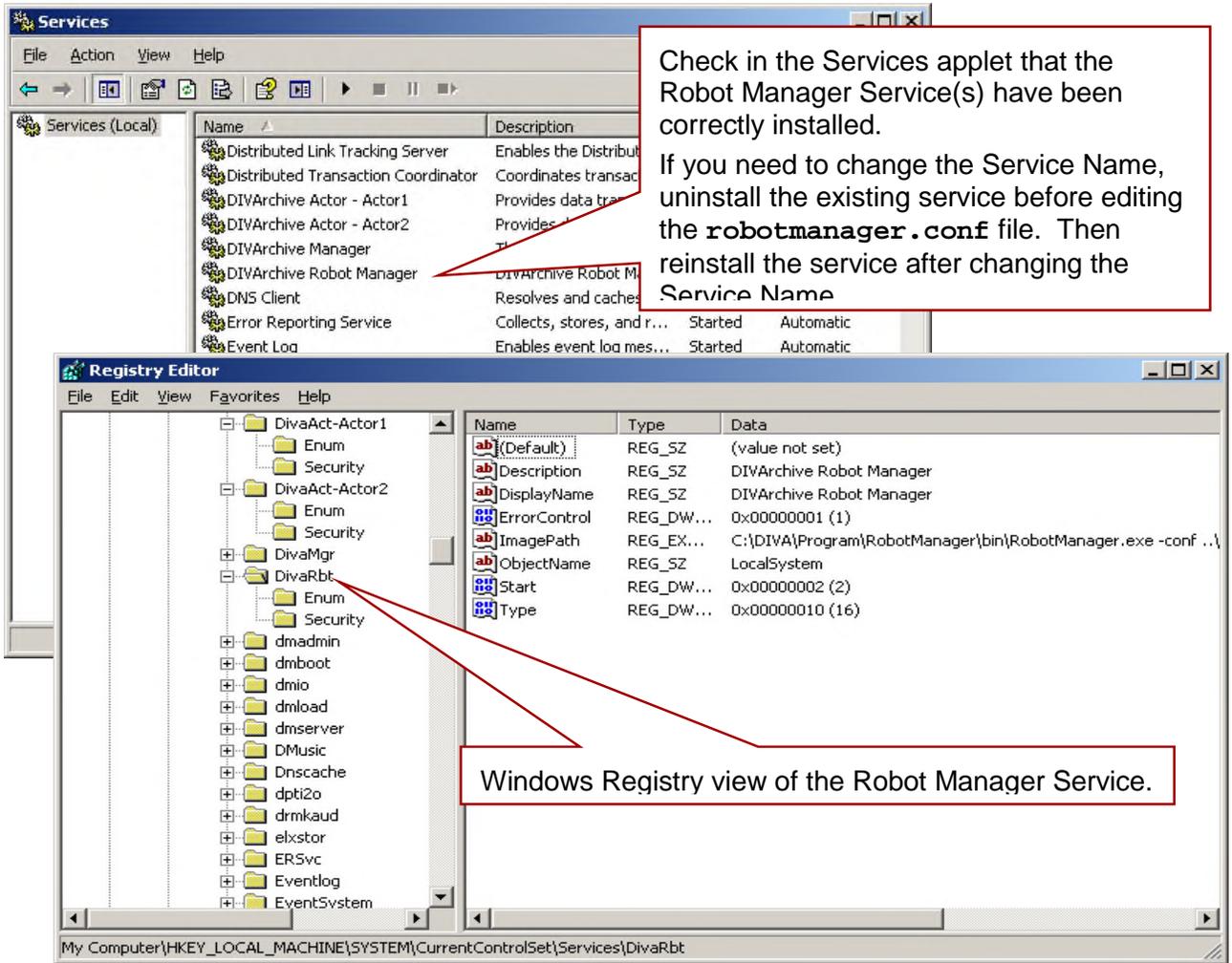
If one or more Robot Manager Services need to be uninstalled, the configuration file path also needs to be specified:

```
robotmanager -u -conf ..\..\conf\robot_manager\robotmanager2.conf
```

Removes the Robot Manager Service as defined by the **service_name** parameter in **robotmanager2.conf**.

After installing the service(s) check the Windows Services applet to make certain that the Robot Manager Service(s) were correctly installed.

Figure 89: Windows Services Applet and Registry Editor



4.9.2 RobotManager Service Management Functions

The following command options are also available for the Robot Manager Service:

- robotmanager debug** Starts the DIVArchive Robot Manager in console mode (*shows diagnostic messages etc. from the library in the console window*).
- robotmanager version (or -v)** Displays the DIVArchive Robot Manager software version information.
- robotmanager help** Displays all command line options.

4.10 Testing the Robot Manager Library Interface

After configuring the Robot Manager Configuration File, launch the DIVArchive Robot Manager and check that the library itself can be controlled.

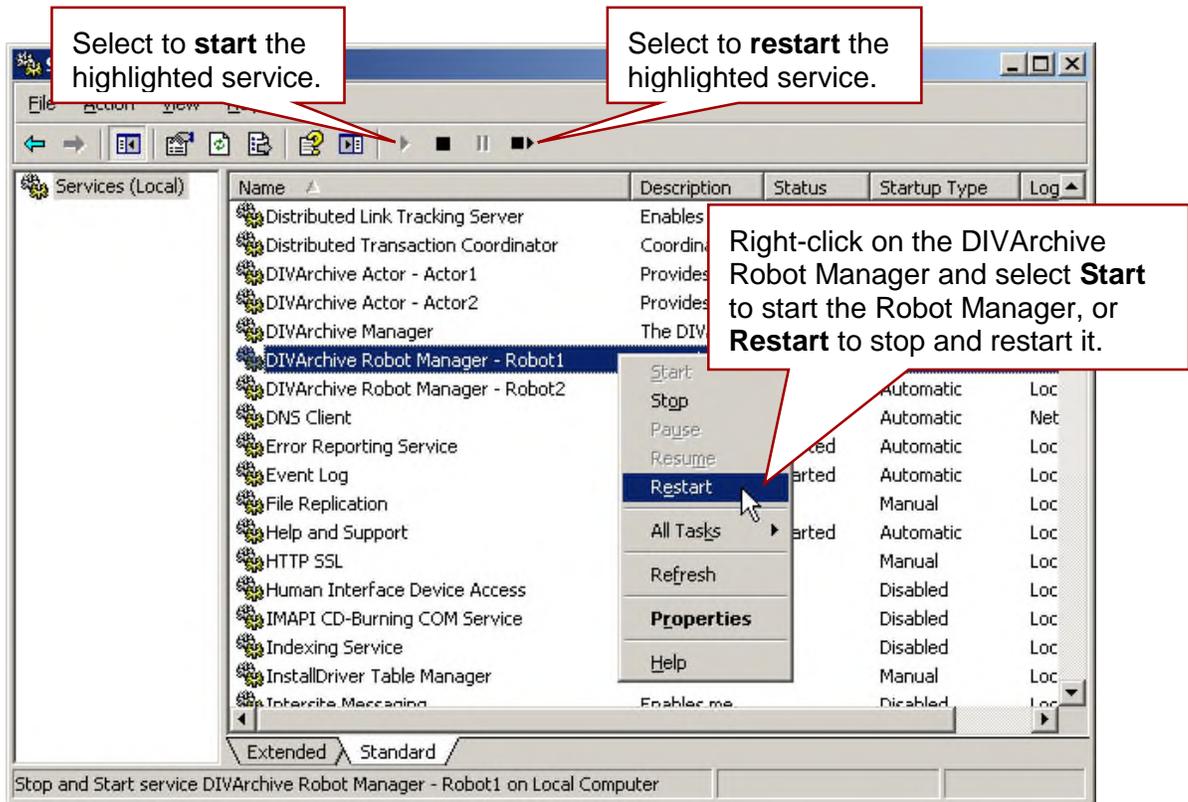
Note: Library interfaces that utilize ACSLS, SDLC, or PSC intermediate control software should be running prior to launching the DIVArchive Robot Manager. ACSLS controlled libraries should also be varied online (e.g. *vary lsm0 online*).

4.10.1 Starting the Robot Manager

Windows DIVArchive Robot Manager(s) start automatically with Windows and can be managed through the Windows Services applet.

Note: If the library is offline when the service is started, the Robot Manager does not automatically reconnect once the library comes online. The service must be restarted in this case.

Figure 90: Starting the Robot Manager in Windows



An alternative way to restart a Robot Manager is to execute the following from a command window (*Start, Run, cmd.exe*).

```
net stop "DIVArchive Robot Manager"  
net start "DIVArchive RobotManager"
```

If the `SERVICE_NAME` has been specified in the `robotmanager.conf` file, then:

```
net stop "DIVArchive Robot Manager SERVICE_NAME"  
net start "DIVArchive RobotManager SERVICE_NAME"
```

Note the quotation marks in the above commands. These **must** be used when specifying a Windows service with spaces in the name.

4.10.2 Testing the Robot Manager Library Control

WARNING: These utilities should NOT be used in a live DIVArchive System. You should not send commands to a Robot Manager via either of these utilities under any circumstances when the DIVArchive Manager is running. Oracle does not accept any responsibility for any complications arising from inappropriate use of these utilities.

The **Robot Manager Client** (*command line interface*) or **GUI** can be used to establish basic control functionality of a Robot Manager to its controlled library(s). Either of these utilities can be used to send manual commands to a DIVArchive Robot Manager to initiate simple operations such as drive mounting, dismounting, enter or eject operations from the Cartridge Access Port (*CAP*). Both utilities connect to a Robot Manager via TCP/IP and can be run from a remote host. This feature allows the Robot Manager GUI to be used from a Windows based remote host.

Note: If you mount a tape with either of these utilities, the tape must be first unloaded before it can be dismounted, unless the library supports Forced Dismount commands and this is enabled in the DIVArchive Robot Manager Configuration File.

4.10.2.1 Robot Manager Client

This command line based client is located with the Robot Manager executable files and normally located in:

- DIVA_HOME\Program\RobotManager\bin.

You need to specify the IP Address of the Robot Manager, and its TCP port when launching this client.

```
RobotManagerClient <IP_Address> <TCP_Port>
```

Where:

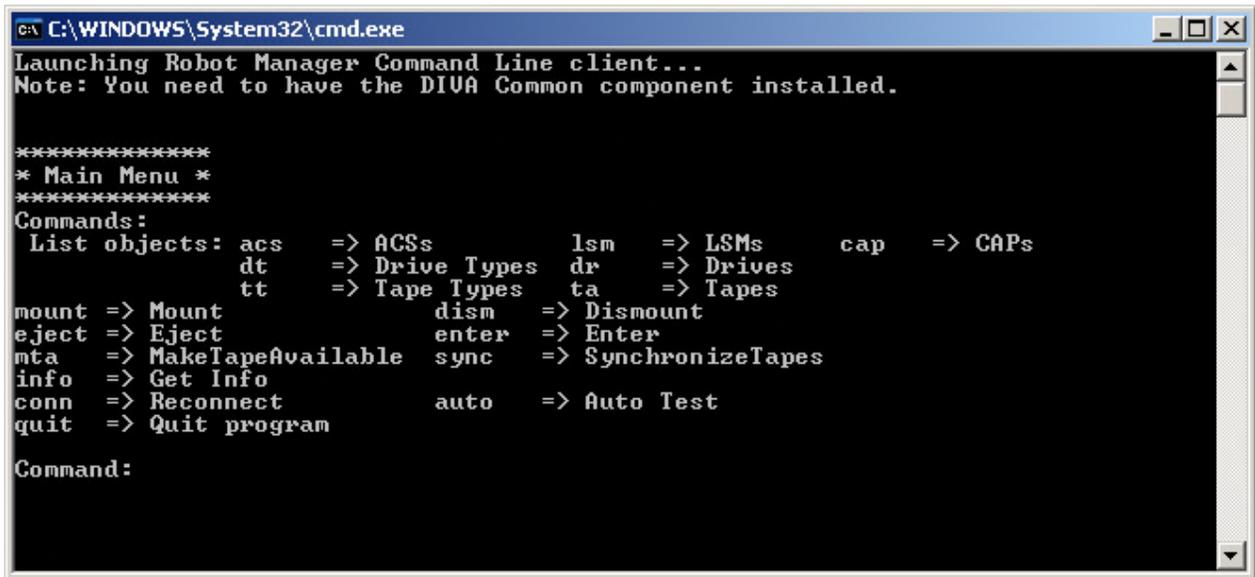
IP_Address The IP Address of the host the Robot Manager is installed on.

TCP_Port The listening port of the Robot Manager.

These two parameters can be hard coded within the Robot Manager Client batch file if there is only one Robot Manager requiring access.

The commands in this client are self-explanatory. Type `quit` to exit the program.

Figure 91: Windows Command Window



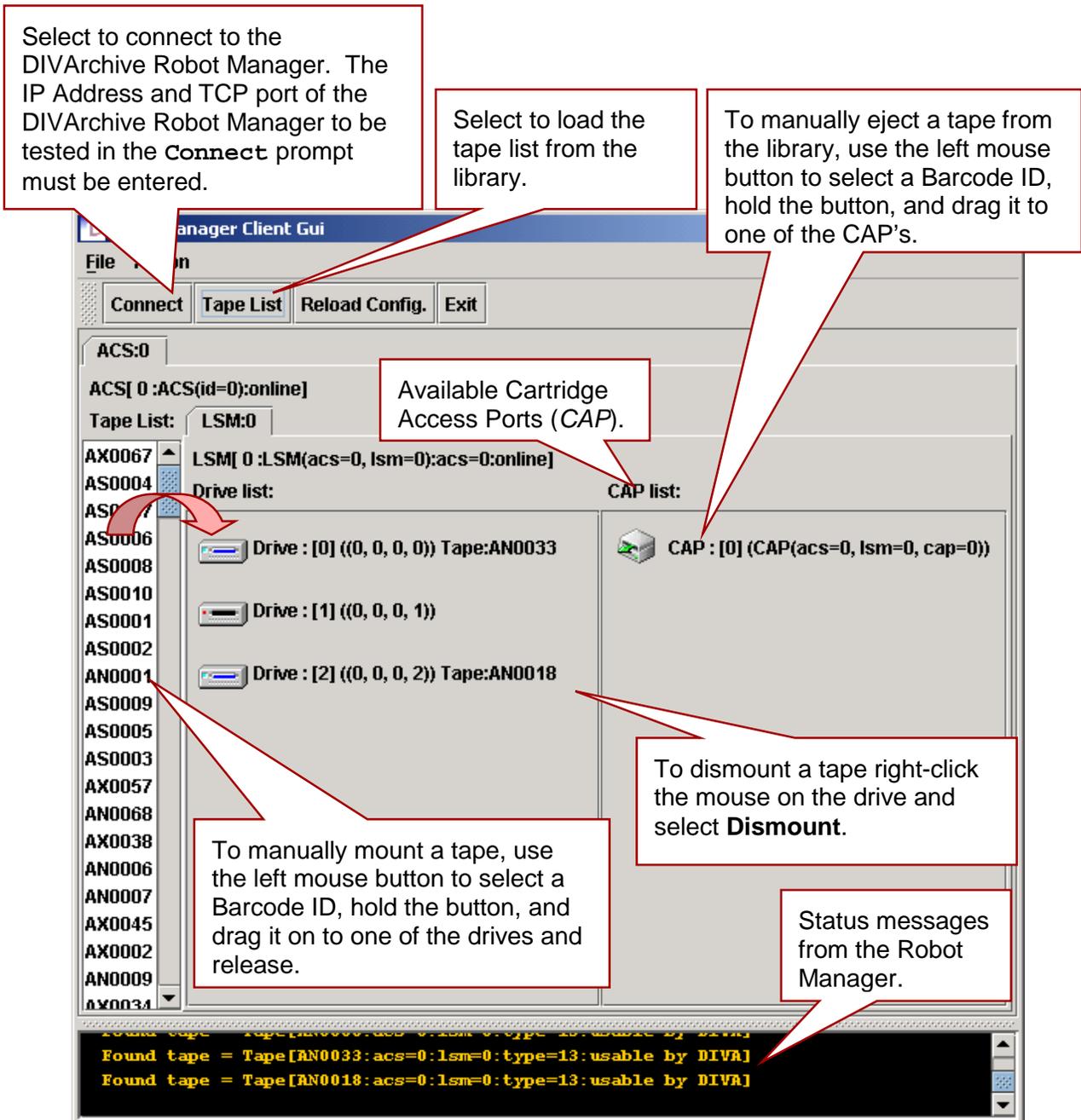
4.10.2.2 Robot Manager GUI

This graphically based client is located with the Robot Manager executable files and normally located in:

- DIVA_HOME\Program\RobotManager\bin.

The GUI provides the same functionality as the command line client and can be launched by running `RobotManagerGUI.bat`.

Figure 92: Robot Manager GUI



4.11 Robot Manager System Level Configuration (DIVArchive Configuration Utility)

4.11.1 Declaring the Robot Manager(s)

At the system level, each instance of the DIVArchive Robot Manager must be declared to the DIVArchive Manager in the **Robots Managers** frame of the **Robots** tab of the Configuration Utility.

Figure 93: Declaring the Robot Manager(s)

The figure shows two screenshots from the DIVArchive Configuration Utility. The top screenshot displays the 'Robot Managers' table with columns for Name, Address, Port, and Site. The bottom screenshot shows the 'Add new row in Robot Managers' dialog box with fields for Name, Address, Port, and Site.

Robot Manager Name	Address	Port	Site
Robotmanager01	192.168.2.100	8500	local
Robotmanager02	172.163.15.2		remote 1
Robotmanager03	202.178.55.21	8500	remote 2

+ Edit - Update

- +** Adds a DIVArchive Robot Manager Host.
- Edit** Edits the selected Robot Manager parameters.
- Removes the selected Robot Manager.

Refreshes the Robot Manager information from the database.

IP Address of the host running the DIVArchive Robot Manager installation.

Name of the DIVArchive Robot Manager attached to this DIVArchive System.

TCP Port of the Robot Manager. This must match the **RM_PORT** parameter specified in **RobotManager.conf**.

Used by the DIVArchive Manager to determine optimal use of resources in resource allocation. Note: Site selection must be enabled in the DIVArchive Manager Configuration File or all sites are considered equally.

4.11.2 Robot Manager – ACS Association

Each DIVArchive Robot Manager is logically referred to by the DIVArchive Manager by its Automatic Cartridge System (ACS) number. This value should be unique amongst all DIVArchive Robot Managers. Individual libraries (or frames) are typically referred to by their Library Storage Module (LSM) number.

Figure 94: Robot Manager – ACS Association

Step 1
Select the **Synchronize DB** option from the Tools menu and acknowledge the warning message.

Step 2
Select the individual Robot Manager to synchronize, or **ALL** to synchronize all Robot Managers.

Step 3
Only check this option.

Step 4
Select **Go** to update.

Step 5
Check for correct operation.

Step 6
Enter details for each library when prompted.

Step 7
Select **Close** to exit

Step 8
Verify Robot Manager – ACS association.

Database Synchronization

Select the robot and synchronization types, then click "Go":
robot1_70, 127.0.0.1, 8501

- Synchronize media types list
- Synchronize tape list (can be very long)
- Synchronize drive types list
- Synchronize drive list
- Synchronize media/drive compatibility list
- Synchronize Robot Manager ACS Associations

Go Cancel Action

Clear Auto-scroll to last line

```
No ACS/Library information available for Robot Manager.  
Library information update from Robot Manager : OK  
Committing the transaction for Robot Manager:- 127.0.0.1:8501  
DB Commit Operation Completed  
Disconnected from the RobotManager  
Operation completed, will perform the database commit.  
DB Commit Operation Completed
```

Library Data Entry

ACS: 0
Serial Number:
Name: ACS_0
Type Description:

Close

Robot Managers-ACS

Robot Manager Name	ACS
robotmanager01	0

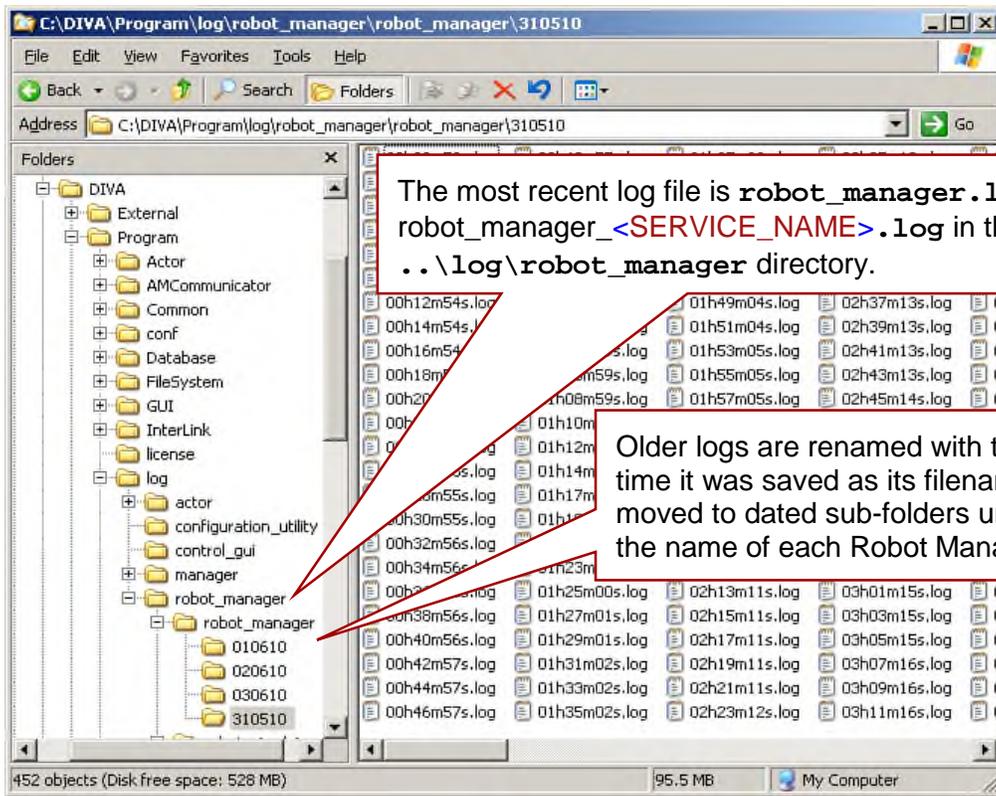
Update

4.12 DIVArchive Robot Manager Logging

4.12.1 Logging Mechanism

During normal operation, each DIVArchive Robot Manager logs communications between itself and the library, and stores them in the `DIVA_HOME\log\robot_manager` folder. These logs are useful in troubleshooting issues with the **Synchronize DB** procedures above or in general with the library. You may be asked to provide these when contacting Oracle Technical Support.

Figure 95: DIVArchive Robot Manager Logging Mechanism



4.12.2 SCSI Error Management

Since the DIVArchive SCSI Robot driver supports multiple library types with a single driver, the error codes provided from each manufacturer's library may differ. If the DIVArchive SCSI Robot Manager encounters an unexpected error from the library after a command is issued, error codes are returned from the library in the following format:

- Sense Key
- Additional Sense Code
- Additional Sense Code Qualifier.

The SCSI Robot Manager will first examine the `scsi_errors.ini` file to match these codes to a specific error message. In order to cater to each library type, library specific configuration files are provided to interpret these error codes correctly if they are not found in `scsi_errors.ini`. When a match is found, it is displayed in the Robot Manager command window and in the Robot Manager logs.

If you have one of the specific library types below, rename the corresponding file to `scsi_specific_errors.ini`.

- `scsi_specific_errors.ini.adic_s1k` For ADIC Scalar 1000 libraries.
- `scsi_specific_errors.ini.adic_s10k` For ADIC Scalar 10000 libraries.
- `scsi_specific_errors.ini.asaca` For Asaca DVD Jukeboxes.
- `scsi_specific_errors.ini.hp-esl` For Hewlett Packard libraries.
- `scsi_specific_errors.ini.ibm-3584` For IBM 3584 libraries.
- `scsi_specific_errors.ini.istora` For ADIC I-2000 libraries.
- `scsi_specific_errors.ini.storagetek` For StorageTek/SUN libraries.

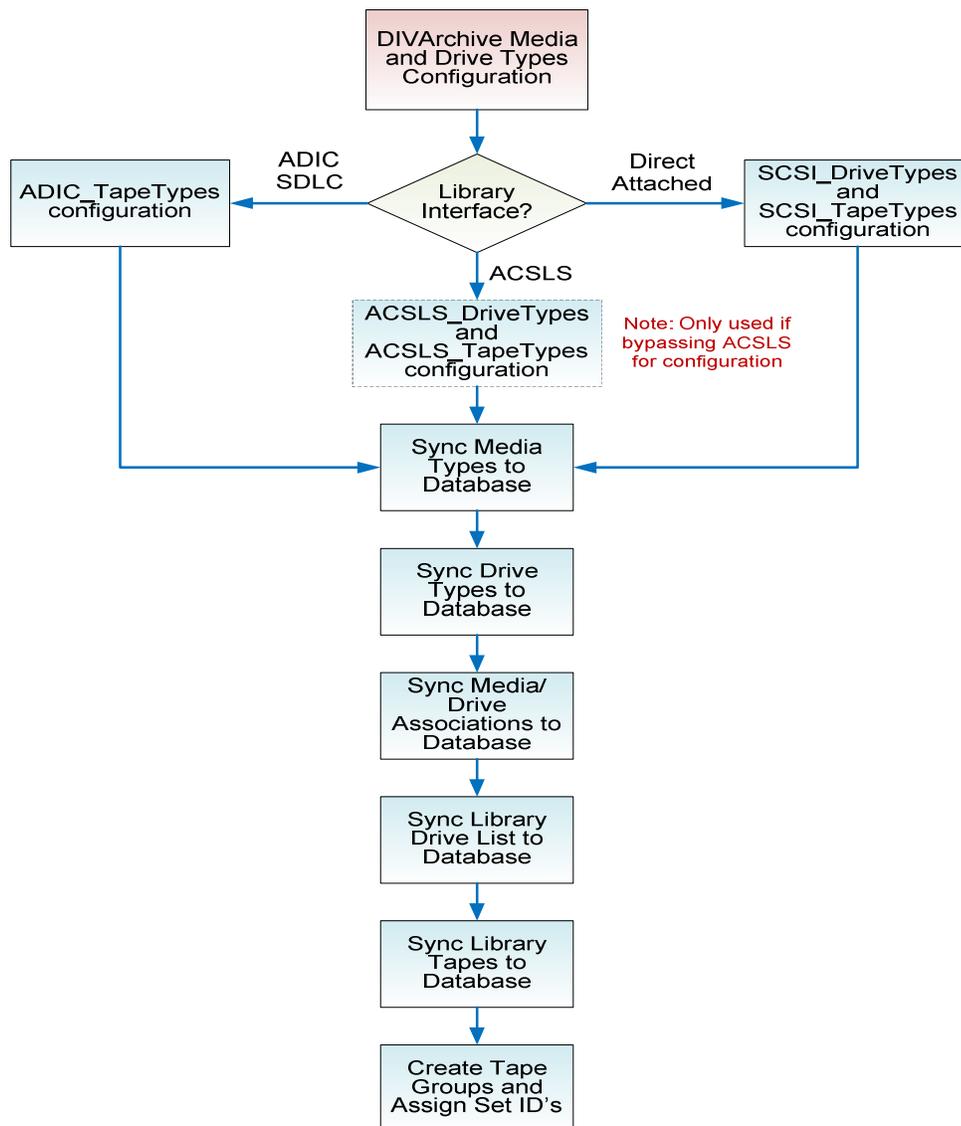
4.13 Media and Drive Types Configuration

Once the DIVArchive Robot Manager has been successfully configured for your library(s), and the appropriate details for all DIVArchive Robot Managers entered into the Robots section of the Configuration Utility, the Tape Media, Drive Models, and the Drive Locations that are currently installed in each library must be entered.

4.13.1 Configuration Overview

The following flowchart lists the workflow of this section of the configuration. All of the DIVArchive Robot Managers configured must be running and successfully connected to each library before commencing this section of the configuration.

Figure 96: Media and Drive Types Configuration Overview



Tape Drives and their associated media types that are installed in a particular library are initially configured in the DIVArchive Database using static configuration files within the `DIVA_HOME\Program\conf\robot_manager` folder. The selection of these files by the DIVArchive Robot Manager depends upon the `RM_MODULE` setting configured in `RobotManager.conf`.

Table 24: Tape Drives and Associated Media

Filename	Usage
<code>scsi_drive_types.ini</code> <code>scsi_tape_types.ini</code>	Direct attached SCSI Libraries. These files will only be considered if the <code>.ini</code> extension is removed.
<code>acsls_drive_types.ini</code> <code>acsls_tape_types.ini</code>	Libraries managed by ACSLS. Normally, Tape and Drive Types are derived from ACSLS during library synchronization with the database; however these files can be used if you need to override the values returned from ACSLS. These files will only be considered if the <code>.ini</code> extension is removed.
<code>adic_media_types.ini</code>	Used in conjunction with ADIC Libraries controlled by SDLC. Drive Types for this library are directly returned from the SDLC server. These files will only be considered if the <code>.ini</code> extension is removed.

When a hardware audit is initiated on the specific library by the Configuration Utility (*through the DIVArchive Robot Manager(s), either directly or via intermediate library management software*), hexadecimal codes are returned to identify the model and order of the tape drives currently installed and the media types present in the library.

The Tape Types and Drive Types configuration files are used to map these library hardware codes to drive and media IDs within the DIVArchive Database.

Note: It is only necessary to modify these files when new Drive or Media Types are added to the library.

4.13.2 SCSI_drive_types / ACSLS_drive_types

These files can be edited with any text editor. No modification of these files is required other than to remove comment fields for the appropriate library and drive types for your installation.

Figure 97: SCSI_drive_types / ACSLS_drive_types

```

# If the SCSI Robot Manager is connected to a StorageTek library
# in direct-attach mode
# above:
# UNCOMMENT ALL LINES IN
#-----
# If the SCSI Robot Mana
# UNCOMMENT THE APPROPRI
# to 0X00. EXAMPLE:
# 47 0X00 0x00
#-----
# TypeID T
#-----
#00 0xFF 0xFF Unknown
#00 0x00 0xFF Unknown
#00 0x01 0xFF Unknown
#00 0x4C 0xFF Unknown
#08 0x01 0x44 DLT7000
#09 0x00 0x02 9840
#11 0x01 0x47 DLT8000
#13 0x00 0x05 T9940A
#16 0x00 0x01 T9940B
#18 0x4C 0x30 HP-LTO
#19 0x4C 0x31 IBM-LTO
#20 0x4C 0x32 SGT-LTO
#21 0x00 0x09 T9940B
#25 0x00 0x0B T9840C
#26 0x00 0x0C T9840C35
#27 0x4C 0x33 HP-LTO-2
#28 0x4C 0x34 IBM-LTO-2
#29 0x4C 0x35 SGT-LTO-2
#31 0x54 0x0D T1A
#32 0x54 0x0E T1A35
#33 0x4C 0x36 HP-LTO-3
#34 0x4C 0x37 IBM-LTO-3
#35 0x4C 0x38 CER-LTO-3
#37 0x54 0x18 T1AE
#38 0x54 0x19 T1AE35
#39 0x00 0x0F T9840CE
#40 0x00 0x11 T9840CE5
#41 0x00 0x12 T9840D

```

DIV Archive Drive ID

Remove the hash (#) symbols in the appropriate library section for the drives to be recognized in a **Synchronize drive types list** in the Configuration Utility.

Library Drive Codes

Drive Types

Compatible Media Types

Cross references Tape Type IDs in **SCSI_Tape_Types** (or **ACSLTape_Types** if used).

These values are examined in the Synchronize Media/Drive Compatibility List procedure in the Configuration Utility.

Drive Types in libraries not installed should be left commented out.

4.13.2.1 SCSI_tape_types / ACSLS_tape_types

These files can be edited with any text editor. No modification of these files is required other than to remove comment fields for the tape types for your specific library.

Figure 98: SCSI_tape_types / ACSLS_tape_types

```
# If the SCSI Robot Manager is connected to a StorageTek library
# in direct-attach mode (no ACSLS server), or to any listed library listed
# above:
# UNCOMMENT ALL LINES IN THIS FILE
#-----
# TypeID TransportDomainID DriveType
#-----
#00 0xFF 0xFF Unknown 08 11
#00 0x00 0xFF Unknown
#00 0x01 0xFF Unknown
#00 0x4C 0xFF Unknown
#00 0x43 0xFF Unknown
#07 0x01 0x00 DLTIV 08 11
#09 0x00 0x52 STK1R 09 16 25
#10 0x00 0x55 STK1U 09 16 25
#13 0x00 0x50 STK2P 13 21
#14 0x00 0x57 STK2W 13 21
#16 0x4C 0x31 LTO-100G 18 19 20 27 28 29 33 34 35
#17 0x4C 0x41 LTO-50G 18 19 20 27 28 29 33 34 35
#18 0x4C 0x42 LTO-35G 18 19 20 27 28 29
#19 0x4C 0x43 LTO-10G 18 19 20 27 28 29
#20 0x43 0x32 LTO-CLN2 19 28 34
#21 0x43 0x33 LTO-CLN3 20 29 35
#22 0x43 0x31 LTO-CLN1 18 19 20 27 28 29 33 34 35 46 47 48
#25 0x43 0x55 LTO-CLNU 18 19 20 27 28 29 33 34 35 46 47 48
#26 0x4C 0x32 LTO-200G 27 28 29 33 34 35 46 47 48
#28 0x54 0x31 T10000T1 31 32 37 38 49 50 51 52 53 54 55 56
#29 0x54 0x53 T10000TS 31 32 37 38 49 50 51 52 53 54 55 56
#30 0x43 0x54 T10000CT 31 32 37 38 49 50 51 52
#31 0x4C 0x33 LTO-400G 33 34 35 46 47 48
#32 0x4C 0x33 LTO-400W 33 34 35 46 47 48
#39 0x00 0x59 STK1Y 41 42 43 44
#40 0x4C 0x34 LTO-800G 46 47 48
#41 0x4C 0x55 LTO-800W 46 47 48
#42 0x54 0x32 T10000T2 53 54 55 56
#43 0x54 0x54 T10000TT 53 54 55 56
#44 0x43 0x43 T10000CC 53 54 55 56
#60 0x4A 0x41 3592-JA 60 61
#61 0x4A 0x42 3592-JB 60 61
#62 0x4A 0x57 3592-JW 60 61
#63 0x4A 0x58 3592-JX 60 61
```

DIVArchive Tape ID
Remove the hash (#) symbols in the appropriate library section for the tapes (or DVD's) to be recognized in a Synchronize Media Types List in the Configuration Utility.

Library Tape Codes

Tape Type

Compatible Drive Types
Cross references Drive Type IDs in SCSI_Drive_Types (or ACSLS_Drive_Types if used).
These values are examined in the Synchronize Media/Drive Compatibility List procedure in the Configuration Utility.
Tape Types (or DVD's) in libraries not installed should be left commented out.

4.13.2.2 ADIC_media_types

These files can be edited with any text editor. No modification of these files is required other than to remove comment fields for relevant tape types for your installation.

Figure 99: ADIC_media_types

```
#####
# If you have an IBM library.
# UNCOMMENT ONE LINE IN THE FOLLOWING
#
# Drive Types Definition. File
# number name [compat_type]
#####
#1 ACI_3480 1
#2 ACI_OD_THICK 2
#3 ACI_OD_THIN 3
#4 ACI_DECDLT 4
#5 ACI_8MM 5
#6 ACI_4MM 6
#7 ACI_D2
#8 ACI_VHS
#9 ACI_3590
#10 ACI_CD 10
#11 ACI_TRAVAN 11
#12 ACI_DTF 12
#13 ACI_BETACAM 13
#14 ACI_AUDIO_TAPE 14
#15 ACI_BETACAML 15
#16 ACI_SONY_AIT 16
#17 ibm-lto 17 49
#18 ACI_DVCM 18
#19 ACI_DVCL 19
#20 ACI_NUMOF_MEDIA 20
#49 ibm-lto 49 17
```

DIVArchive Tape ID
Remove the hash (#) symbol for the tape to be recognized in a Synchronize Media Types List in the Configuration Utility.

Tape Type

Compatible Drive Types
Cross references Drive Type IDs that are returned from the SDLC controller. These values are examined in the Synchronize Media/Drive Compatibility List procedure in the Configuration Utility. Tape Types not installed should be left commented out.

4.14 Synchronizing Media Types with the DIVArchive Database

The values that have been uncommented in the `Tape_Types` configuration files must also be imported into the DIVArchive Database with the following procedure. For successful completion of this procedure, each DIVArchive Robot Manager to be queried should be online.

Note: Only perform this operation if you are adding additional Media Types to the library.

Figure 100: Synchronizing Media Types with the DIVArchive Database

Step 1
Select the **Synchronize DB** option from the Tools menu and acknowledge the warning message.

Step 2
Select the individual DIVArchive Robot Manager to synchronize, or **ALL** to synchronize all Robot Managers.

Step 3
Only check this option.

Step 4
Select **Go** to update. The Configuration Utility will connect to the DIVArchive Robot Manager, which will in turn parse the *SCSI_Tape_Types* (or *ACSL5_Tape_Types* if used) configuration file.

Step 5
If a Tape Type is not currently in the database, you will be prompted to enter it. Say **No** to any Tape Types not currently in use.

Step 6
Enter the Total Size and Block Size for this Media Type. Ensure the Block size is entered correctly before clicking ok as you cannot change it afterwards.

Note: Cleaning tapes (if reported) should be entered with a Tape Size and Block Size of 1KB so they do not interfere with the total available size computation of all tapes in the Control GUI.

Step 7
Select **Close** on the Database Synchronization window and confirm the Tape Type has been correctly entered in the Tape Properties section of the Configuration Utility Tapes tab.

Media Type	Total Size	Block Size
LTO-100G	1,000,000	65,536
LTO2-100G	1,000,000	65,536

4.15 Synchronizing Drive Types with the DIVArchive Database

The values that have been uncommented in the `Drive_Types` configuration files must also be imported into the DIVArchive Database with the following procedure. For successful completion of this procedure, each DIVArchive Robot Manager to be queried should be online.

Note: Only perform this operation if you are adding additional Drive Types to the library.

Figure 101: Synchronizing Drive Types with the DIVArchive Database

Step 1
Select the **Synchronize DB** option from the Tools menu and acknowledge the warning message.

Step 2
Select the individual DIVArchive Robot Manager to synchronize, or **ALL** to synchronize all Robot Managers.

Step 3
Only check this option.

Step 4
Select **Go** to update. The Configuration Utility will connect to the Robot Manager, which will in turn parse the *SCSI_Drive_Types* (or *ACSL_S_Drive_Types* if used) configuration file.

Step 5
If the Drive Type is not currently in the database, you will be prompted to enter it.

Step 6
Enter the Block Size for this drive type. Ensure the Block size is entered correctly before clicking **OK** as you cannot change it afterwards.

Step 7
Check there are no errors in the status window. If so, recheck the Tape Types and Drive Types definition files.

Step 8
Select Close on the Database Synchronization window and confirm the Drive Type has been correctly entered in the Drive Properties frame of the Configuration Utility Drives tab.

Database Synchronization

Select the robot and synchronization types, then click "Go":

All

Synchronize media types list
 Synchronize tape list (can be very long)
 Synchronize drive types list
 Synchronize drive list
 Synchronize media/drive compatibility list
 Synchronize Robot Manager ACS Associations

Go Cancel Action

Auto-scroll to last line

```

Connecting to the Robot Manager on 127.0.0.1/8501...
Connected to Robot Manager on 127.0.0.1/8501

Updating drive types ...
Loading drive properties from DB...
Found in DB : Drive Properties Info : _dpBlockSize "65536", _dpDriveType "simu_IBM_LTO"
Found in DB : Drive Properties Info : _dpBlockSize "1000000", _dpDriveType "simu_IBM_LTO2"
Loading Drive Types from RobotManager ...
Drive Type returned by RobotManager : RDriveType[ 109 :simu_IBM_LTO]
Drive Type found in DB : _dpBlockSize "65536", _dpDriveType "simu_IBM_LTO"
Nothing to do
Drive Type returned by RobotManager : RDriveType[ 110 :simu_IBM_LTO2]
Drive Type found in DB : _dpBlockSize "1000000", _dpDriveType "simu_IBM_LTO2"
Nothing to do

Robot Manager:- 127.0.0.1:8501

```

Question

Do you want to take into account in the database the drive type 109(simu_IBM_LTO)?

Yes No

Enter Block Size

Enter the Block Size (Bytes) of the drive type:
 109(simu_IBM_LTO):
 65536

OK

Drive Properties

Drive Type	Drive Type Id
simu_IBM_LTO	109
simu_IBM_LTO2	110

4.16 Tape Capacity and Block Size Definitions

These values should be used when entering any new Drive or Media Type to the DIVArchive Database. The values have been tuned by Oracle to avoid tape spanning and therefore may be lower than the theoretical capacity.

Table 25: Tape Capacities

Media Type	Drive Type	Capacity
9840	STK 9840A/B	19 531 008
	STK 9840C	39 062 272
9940	STK T9940A	58 593 536
	STK T9940B	195 312 384
T10000T1	STK T10000A	488 281 008
	STK T10000B	976 562 176
T10000TS	STK T10000A	117 187 072
	STK T10000B	234 374 656
	STK T10000C	5 243 000 000
DTF-2	GY-8240	195 312 448
SAIT1	S-AIT 1	488 281 088
SAIT2	S-AIT 2	781 249 536
AIT3	AIT 3	97 656 192
LTO-5	IBM/HP	1 464 843 264
LTO-6	IBM/HP	2 441 405 952
LTO-100G	IBM/HP/Seagate LTO Ultrium 1	97 656 192
LTO-200G	IBM LTO Ultrium 2	195 312 128
LTO-400G	IBM/HP LTO Ultrium 3	390 624 768
LTO-800G	IBM/HP Ultrium 4	781 249 536
DLT-IV	Quantum DLT7000	34 179 648

Table 26: Tape Block Sizes

Manufacturer	Tape Drive Type	Block Size in Bytes
SUN (StorageTek)	T9840A/B/C	262144
	T9940A/B	262144
	T10000A/B/C	524288
Sony	GY-8240 (DTF-2)	65536
	AIT-3	65536
	S-AIT 1	524288
	S-AIT 2	262144
IBM	LTO Ultrium 1	65536
	LTO Ultrium 2/3/4	524288
HP	LTO Ultrium 1	65536
	LTO Ultrium 2/3/4	524288
Seagate	LTO Ultrium 1	65536
Quantum	DLT 7000	65536

4.17 Synchronizing Media and Drive Compatibility with the DIVArchive Database

This step cross-references the compatibility entries in the `Tape_Types` and `Drive_Types` Definition Files. This procedure should only be performed whenever a new Media or Drive Type is added, or to take into account updates from the Tape or Drive Types Definition Files in a DIVArchive software update.

For libraries controlled by ACSLS Media and Drive Type information is normally retrieved directly from ACSLS, hence an ACSLS software upgrade or a library firmware update may require the Media and Drive Type settings to be resynchronized.

Figure 102: Synchronizing Media and Drive Compatibility with the DIVArchive Database

Step 1
Select the **Synchronize DB** option from the Tools menu and acknowledge the warning message.

Step 2
Select the individual DIVArchive Robot Manager to synchronize, or **ALL** to synchronize all Robot Managers.

Step 3
Only check this option.

Step 4
Select **Go** to update. The Configuration Utility will connect to the selected Robot Manager, which will in turn parse the `SCSI_Tape_Types` (or `ACSLTape_Types` if it is being used) configuration files.

Step 5
Ensure there are no errors in the status window. If so, recheck the `Tape_Types` and `Drive_Types` Definition Files.

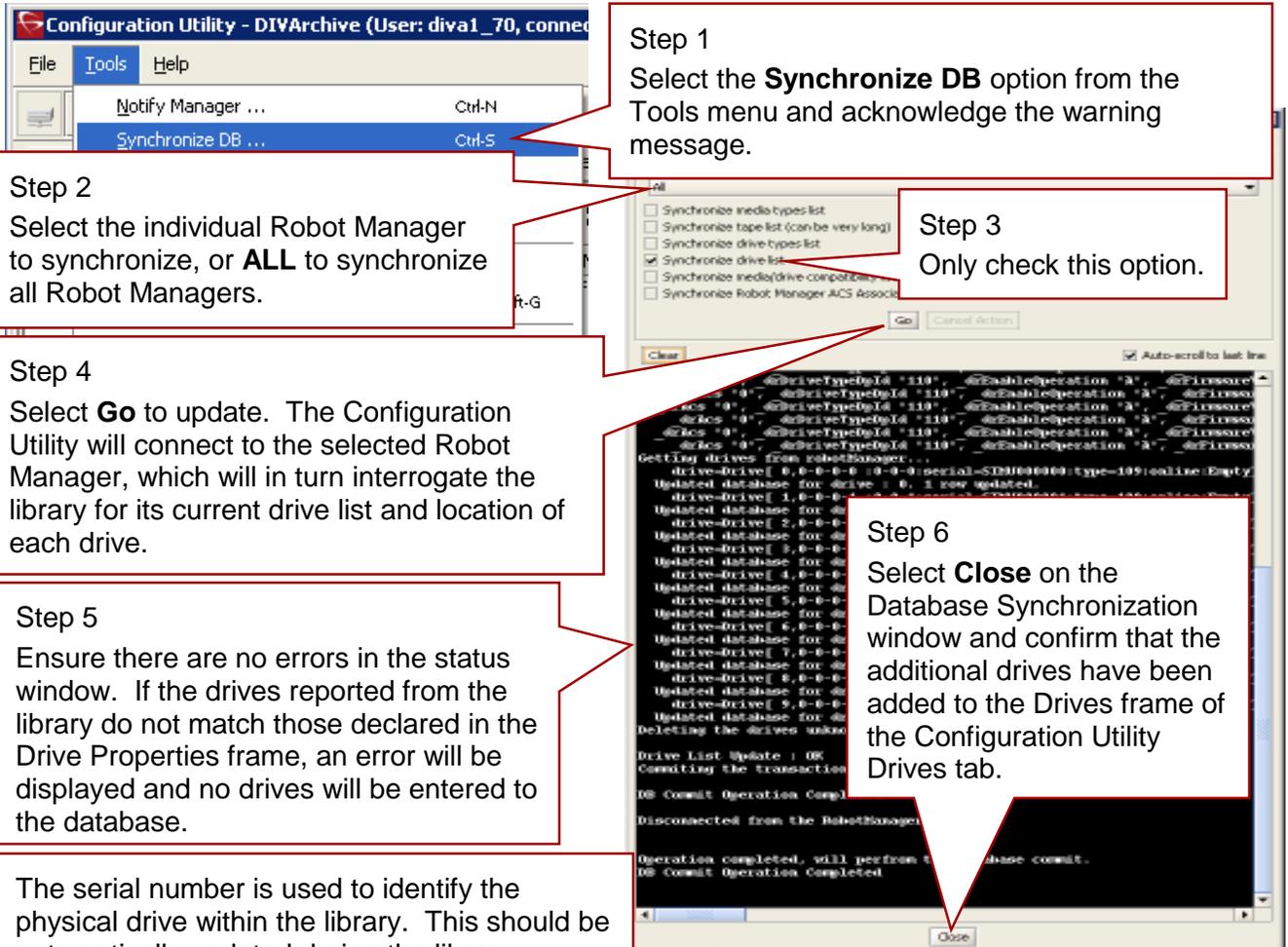
Step 6
Select **Close** on the Database Synchronization window and confirm the Media and Drive Type association has been correctly entered in Media Compatibility frame of the Configuration Utility Robots tab.

Media Type ID	Drive Type ID	Compatibility Type
LTO-100G	simu_IBM_LTO	READ-WRITE
LTO2-100G	simu_IBM_LTO2	READ-WRITE

4.18 Synchronizing the Library Drive List with the DIVArchive Database

If new Drive Types, or additional drives, are added to a DIVArchive Managed Library, they must be declared in the DIVArchive Database. To add the drives to the database, perform the following procedure. For more information on assignment of Drives to Actors refer to Section 5.11.

Figure 103: Synchronizing the Library Drive List with the DIVArchive Database



Step 1

Select the **Synchronize DB** option from the Tools menu and acknowledge the warning message.

Step 2

Select the individual Robot Manager to synchronize, or **ALL** to synchronize all Robot Managers.

Step 3

Only check this option.

Step 4

Select **Go** to update. The Configuration Utility will connect to the selected Robot Manager, which will in turn interrogate the library for its current drive list and location of each drive.

Step 5

Ensure there are no errors in the status window. If the drives reported from the library do not match those declared in the Drive Properties frame, an error will be displayed and no drives will be entered to the database.

Step 6

Select **Close** on the Database Synchronization window and confirm that the additional drives have been added to the Drives frame of the Configuration Utility Drives tab.

The serial number is used to identify the physical drive within the library. This should be automatically updated during the library synchronization. Refer to Section 4.19 if this information needs to be entered manually.

When this field is set to **N**, the drive is ignored by DIVArchive (*and will not be displayed in the Control GUI Drives tab*). If you subsequently set a drive to **Y**, DIVArchive will not use it until the Manager has been notified. This field is used to restrict drive usage in libraries that are shared with other backup or archive applications.

Drives

ID	Serial Number	Firmware Version	ACS	LSM	Drive Type	Status	Operations	Used	Inter
0	SIMU000000	79M0	0	0	109	Online	A	Y	0-0-0
1	SIMU000001	79M0	0	0	109	Online	A	Y	0-0-1
2	SIMU000002	79M0	0	0	109	Online	A	Y	0-0-2
3	SIMU000003	79M0	0	0	109	Online	A	Y	0-0-3
4	SIMU000004	79M0	0	0	109	Online	A	Y	0-0-4

Library Drive ID

New drives that are added are initially set **Offline** and thus disabled. Before they can be used, they must be set to **Online** and the Manager (*if running*) notified. During DIVArchive operations, the Manager may automatically set a drive to **Offline** if it encounters a problem with it.

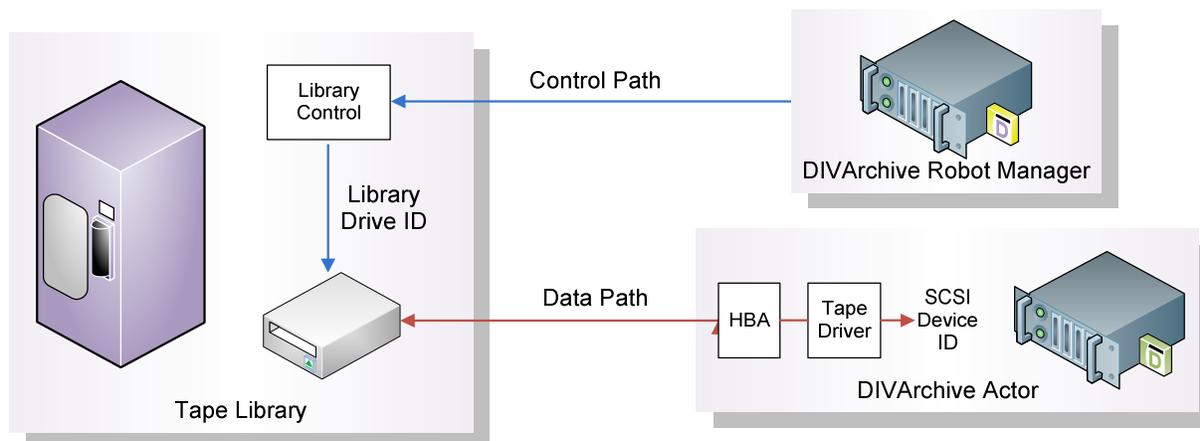
Defines which operations are permitted on each drive:

- R** Repack Only – Dedicates this drive to Tape Repacking **only**.
- S** Standard Functions -All operations **except** Tape Repack.
- A** All Functions - Standard and Repack operations.
- N** No Operations – The drive will not be used (it can, however, later be enabled without a Manager restart).

4.19 Identifying Drive Serial Numbers Manually

When utilizing a Tape Library with DIVArchive there are essentially two logical connections to each drive in that library. The first is the Robotics Control (*handled by the DIVArchive Robot Manager*) for mounting and dismounting the tapes from specific drives, and the Data Interface to the drive itself from the Actors. The figure below illustrates these two connection paths.

Figure 104: Identifying Drive Serial Numbers Manually



Tape libraries identify their drives by the **Drive ID** (*typically 0, 1, 2, etc.*). When the DIVArchive Robot Manager instructs the library to mount a tape to a specific Drive ID, DIVArchive also needs to know the corresponding data path to that drive from each Actor. If the Actor/Library mapping is incorrect, the result will be DIVArchive attempting to read or write to the wrong drive (*resulting in possible data loss or corruption*).

The host operating system presents each drive to applications via their SCSI IDs that can vary as hardware is added or removed, and particularly when shared amongst multiple hosts in a SAN based environment. This configuration would require statically configured SCSI IDs via persistent bindings, which greatly complicates drive replacement.

To simplify configuration and streamline future drive replacements, the data path mapping to each drive with respect to its physical location in the library is achieved by utilizing its unique serial number rather than its SCSI ID. When a DIVArchive Actor is launched, it interrogates each drive's serial number and compares it to the values in the database. It then establishes the correct data path to the drive, irrespective of its SCSI ID.

Each drive's serial number is automatically identified via library synchronization through the Configuration Utility during initial installation or drive replacement. Some cases may need the serial number to be determined manually and entered into, or verified against, the database.

The drive serial number can be manually determined by either using the front panel display of the library itself, or using the Scandrive Utility in conjunction with the Robot Manager Client or GUI.

The latter method involves mounting a tape into a specific drive number in the tape library, establishing which drive the Actor is reporting that it has that tape mounted, and then recording its serial number and entering or verifying it against the corresponding library

Drive ID in the database. This process only needs to be completed once for each drive in the library.

Warning: The following procedures directly interact with both the Robot Manager(s), and the Tape Drives in the library(s). Therefore, they should NOT be attempted while the DIVArchive Manager is running.

4.19.1 Mounting/Dismounting a Tape

The Robot Manager GUI client utility is located in the `DIVA_HOME\Program\RobotManager\bin` folder and can be used to send manual mount commands to a DIVArchive Robot Manager. Launch the utility by executing `RobotManagerGUI.bat`.

This utility is used to issue direct commands to the Robot Manager(s), and **will** interfere with DIVArchive operations. It should **NOT** be used while the DIVArchive Manager is running.

With all drives empty, drag a tape from the tape list column to one of the listed drives. The library should now begin to load a tape into the selected drive. Check the library itself to verify that the correct barcoded tape has now been loaded to the physical drive selected.

Figure 105: Mounting/Dismounting a Tape

The screenshot shows the 'Manager Client Gui' window with a menu bar (File, Action) and buttons for 'Connect', 'Tape List', 'Reload Config.', and 'Exit'. The main area displays 'ACS:0' and 'ACS[0 :ACS(id=0):online]'. Below this is the 'Tape List' section for 'LSM:0', showing a list of drives and their corresponding tapes. A 'CAP list' section shows 'CAP : [0] (CAP(acs=0, lsm=0, cap=0))'. A status bar at the bottom shows messages like 'Found tape = Tape[AN0033:acs=0:lsm=0:type=13:usable by DIVA]'. Red callout boxes provide instructions: 'Select to connect to a DIVArchive Robot Manager...', 'Select to load the Tape List from the library.', 'To manually eject a tape from the library use the left mouse button to select a Barcode ID...', 'Available Cartridge Access Ports (CAP).', 'To manually load a tape use the left mouse button to select a Barcode ID...', 'To dismount a tape, right-click the mouse on the drive, and select Dismount.', and 'Status messages from the Robot Manager.'

The serial number of each drive can be discovered by using the `scandrive.exe` utility located in the `DIVA_HOME\Program\Actor\bin` folder. This utility automatically reports all SCSI devices installed in the Windows host, and their corresponding *port*, *bus*, *target* and *logical unit numbers*. For tape devices, the utility also indicates the drive's firmware, serial number, and whether or not a tape is loaded into each drive.

After a tape is mounted in a drive (using the Robot Manager GUI client), run the `scandrive.exe` utility on an Actor Host (that will use the selected drive) to determine which drive has the tape mounted and its corresponding serial number.

Figure 106: Running Scandrive Utility on Actor Host

```

C:\>cmd.exe - scandrive
C:\DIUA\Program\Actor\bin>scandrive
Scsi3:0:1:0
  Type : TapePeripheral
  Identifier : IBM      ULTRIUM-TD3      64D0
  DeviceName : Tape1
  Serial number : 1210092713
  Firmware version : 64D0
  Mounted tape : 000152.

Scsi3:0:2:0
  Type : TapePeripheral
  Identifier : SONY     SDZ-130
  DeviceName : Tape2
  Serial number : 0001200317
  Firmware version : 0202
  Status : No media in drive.

Scsi3:0:3:0
  Type : TapePeripheral
  Identifier : SONY     SDZ-230      0100
  DeviceName : Tape3
  Serial number : 0001468132
  Firmware version : 0100
  Status : No media in drive.

Press Enter to exit . . .
  
```

The **Type** section refers to that peripheral's class (*HDD, CDROM etc.*). Each tape drive will be reported as a **TapePeripheral**, and the **Identifier** for each corresponding device should match the model number of the drive itself (e.g. *IBM Ultrium TD2*).

Confirm this is the tape loaded via the Robot Manager Client GUI. The serial number should then be entered to the corresponding library Drive ID in the Configuration Utility.

Remember to unmount the tape after determining the drive's serial number.

Drive Serial Number and Firmware version.

ID	Serial Number	Firmware Version	ACS	LSM	Drive Type	Status	Operations	Used	Internal Name
0	SIMU000000	79M0	0	0	109	Online	A	Y	0-0-0
1	SIMU000001	79M0	0	0	109	Online	A	Y	0-0-1
2	SIMU000002	79M0	0	0	109	Online	A	Y	0-0-2
3	SIMU000003	79M0	0	0	109	Online	A	Y	0-0-3
4	SIMU000004	79M0	0	0	109	Online	A	Y	0-0-4

Enter each serial number for the appropriate drive by highlighting it in the **Drives** section of the Configuration Utility, and selecting the **Edit** button.

Repeat the above process by mounting a tape into the next library drive.

4.20 Synchronizing the Library Tapes with the DIVArchive Database

Each tape that is inserted into a library is initially identified by its barcode label. DIVArchive keeps a track of tapes that are currently in the library or that have been removed (*externalized*) in its database.

The labels and status (*whether internalized or externalized*) are updated in the database by `Insert Tape` or `Eject Tape` commands that are issued to DIVArchive. The database can get un-synchronized with a library's contents when tapes are added or removed directly in the library rather than through DIVArchive. Use this procedure to re-synchronize the tape list in the database with the library contents.

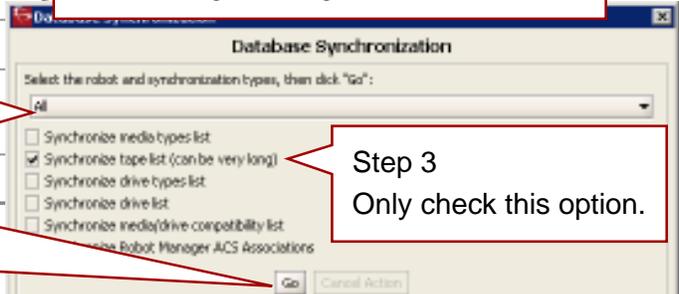
Tip: This procedure is also a quick way to populate the database with tapes from the library when it is initially first loaded with tapes.

Figure 107: Synchronizing the Library Tapes with the DIVArchive Database



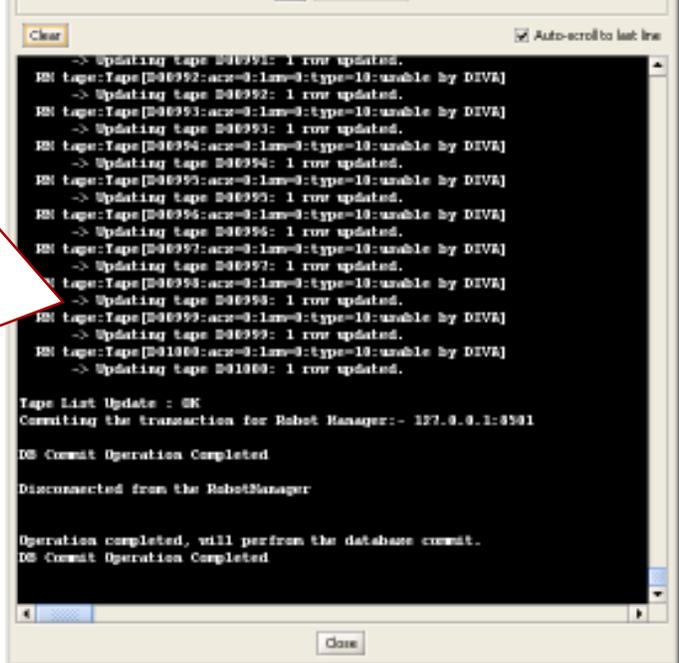
Step 1
Select the **Synchronize DB** option from the Tools menu and acknowledge the warning message.

Step 2
Select the individual DIVArchive Robot Manager to synchronize, or **ALL** to synchronize all Robot Managers.



Step 3
Only check this option.

Step 4
Select **Go** to update. The Configuration Utility will connect to the Robot Manager, which will in turn interrogate the library for its current list of tapes.



Step 5
Tapes in the library are compared to the tape tables in the DIVArchive Database. New tapes are inserted into the table and existing tapes have their status updated (*internalized or externalized*).
If a Tape Type is reported that does not match the types configured in the Tape Properties frame (4.14), an error will be reported and no update of the database will occur. This type of error may also occur if a library is unable to correctly read a tape's barcode label (*carefully check the Robot Manager logs in this case*).

Step 6
Select **Close** to finish. New tapes are added to the Unused Tape Sets frame in the Sets, Groups, and Media Mapping tab of the Configuration Utility and assigned to Set ID 1. Tapes currently tracked by DIVArchive that are found to be missing in the audit will have their status updated to **externalized**. The status of all tapes can be examined in the DIVArchive Control GUI.

Unused Tapes Sets					Edit	Refresh
Barcode	ACS	LSM	Media Type	Set ID		
D00002	0	0	ID=9	1		
D00003	0	0	ID=9	1		
D00004	0	0	ID=9	1		
D00005	0	0	ID=9	1		
nnnnn6	n	n	tn=a	1		

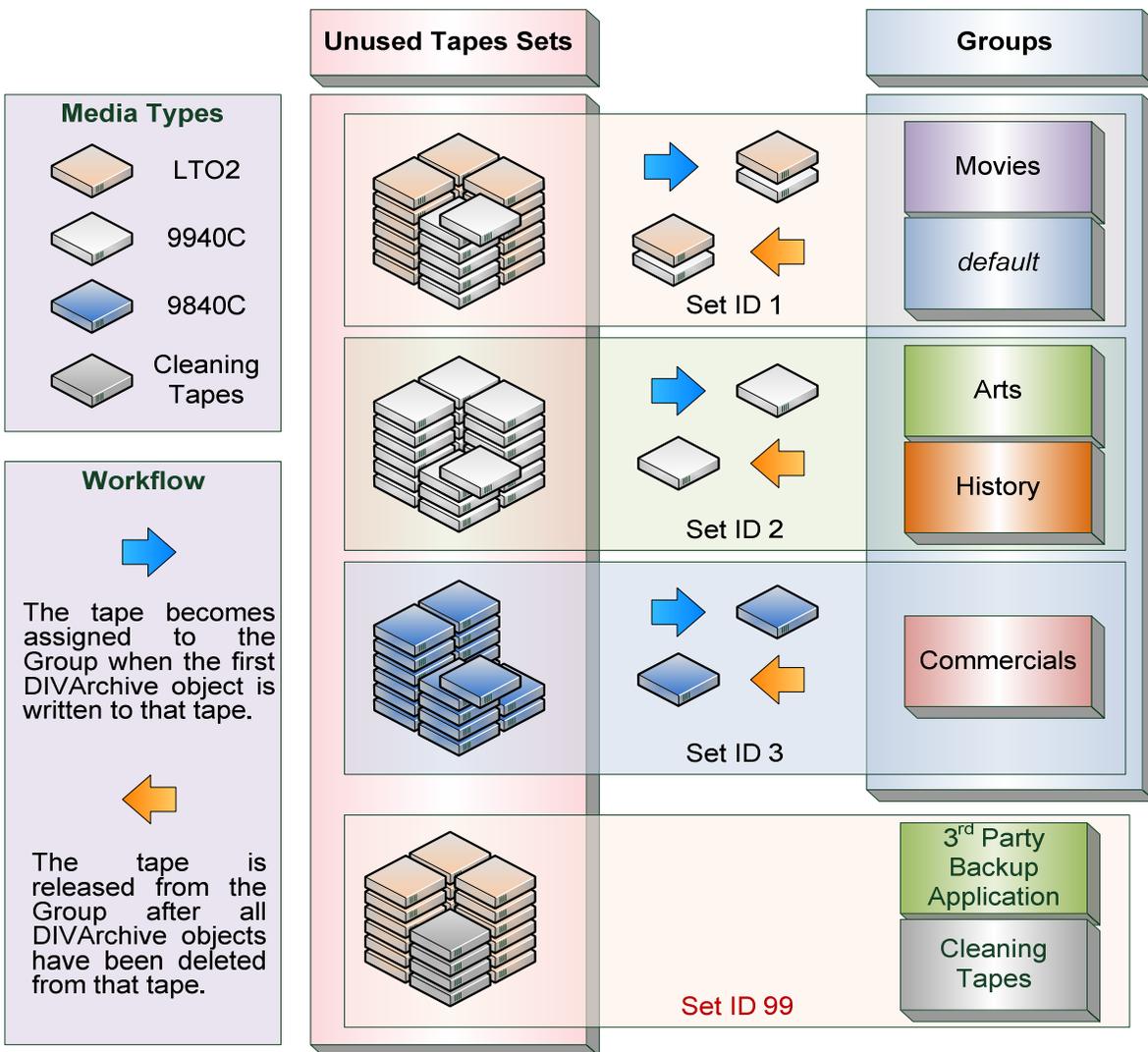
4.21 Creating Tape Groups

The **Sets, Groups, & Media Mapping** tab of the Configuration Utility is used to define Tape Groups within the archive. Groups are used to segment material within the tape library or to associate content with a particular media type. The default group is present in all installations and cannot be removed; however, it is not necessary to use it if you wish to specify your own Group Names. Generally, the Group Name is descriptive of the function or content that is being stored.

A group is associated with a **Set ID** which defines the pool of tapes it can draw upon to store DIVArchive Objects. A tape from the pool is only assigned to a group when an object is written to it, and then released from the group when all objects have been deleted or the tape has been repacked.

The following diagram illustrates these concepts.

Figure 108: Creating Tape Groups



The group concept in combination with the Set ID allows optimal use of tape resources. Some tape drives and media are extremely fast but typically have less storage than their larger capacity (*and slower*) counterparts. Content that is small, or required very quickly, would be archived to this group and should use the faster drives.

For example, the SUN 9840C Tape Drive is (*by today's standards*) small in capacity, however provides extremely fast access times (*approximately 15 seconds from mount to data retrieval*), and is much better suited to storing large numbers of relatively small data files (*particularly in relation to tape repacking*). In this example, the Commercials Group is allocated the Set ID of 3 and all 9840C tapes assigned to that set. Short form commercial material written to tape will exclusively use the 9840C media. Longer (*and larger*) material, such as movies and interstitial programs are better suited to the larger capacity tape sets.

Figure 109: Creating Tape Groups in DIVArchive

The screenshot shows the 'Group' management interface. At the top, there are buttons for '+', 'Edit', '-', and 'Refresh'. Below is a table of existing groups:

Id	Group Name	Set ID	Description	Media Types	Tape Format	Worse Fit Enable
10	leg_vw	1			Legacy	
3	axf	1			AXF	N
1	default	1	The default mandatory group		AXF	N
					Legacy	N

An 'Add new row in Groups' dialog box is open, showing the following fields:

- Id:** 21
- Group Name:** (empty)
- Set ID:** 1
- Description:** (empty)
- Tape Format:** AXF
- Worse Fit Enable:** Y
- Repack Reservation:** 0
- Verify Write(VW):** OFF
- Default Checksum Type:** MD5

Callouts provide the following information:

- Group Names:** These names will appear in the **MEDIA** drop-down selection of an archive request in the Control GUI.
- Default Set ID of each group is 1.** Tapes cannot be assigned to additional Set IDs until they are first created in a group.
- Select + to add a new group.** A group name cannot be edited or deleted if it already contains DIVArchive objects. (A Group can also be added or removed through the DIVArchive Client API from a 3rd party archive initiator).
- Tape Media Types currently in use by this group.** This is updated automatically when a tape is assigned to the groups Set ID in the **Unused Tapes Sets** frame.
- Name of the group as it will appear in the MEDIA tab in the Control GUI.**
- Description of the group is arbitrary.**
- Use the pull-down box to select the Tape Format.**
- This group's Set ID.** Tapes assigned to this Set ID will be available to this group. Note: Set ID 0 is a special set reserved for internal DIVArchive processing and should never be used.
- This only applies if the worst fit option is enabled.** It sets the number of unused tapes in the pool to reserve for tape repacking. All other groups that also use this group's Set ID should also have identical values.
- DIVArchive by default attempts to fill any tapes already assigned to a group prior to assigning an unused tape.** The Worst Fit Option attempts to span objects on as many tapes as possible.

4.22 Creating Tape Sets

When a new tape is entered into a library or DIVArchive clears a tape of its objects (*i.e. all objects on that tape have been deleted, have been migrated to another tape, or moved to another tape after a tape repack*), the tape is released back to the **Unused Tapes Sets** pool.

By default, new tapes are automatically assigned a Set ID of 1, which is the default in all DIVArchive installations. Other Set ID numbers are typically used to distinguish between different types of media but could be used to create restricted pools of tapes for particular applications. If this is the case in your installation, the Set ID must be updated for these tapes after they are inserted into the library.

Figure 110: Creating Tape Sets

The ACS/LSM number is the specific library where the tape is located.

Edits a tape's Set ID: Select multiple tapes by selecting a row with the mouse and pressing **CTL** or **SHIFT** to select a range.

Refreshes this table listing from the DIVArchive Database.

Tapes not currently in use by a **Group**.

Barcode	ACS	LSM	Media Type	Set ID
D00002	0	0	ID=9	1
D00003	0	0	ID=9	1
D00004	0	0	ID=9	1
D00005	0	0	ID=9	1
D00006	0	0	ID=9	1

Select the Set ID for the tape from the drop-down menu (Only Set IDs that have been created first in the Groups window will be listed).

Setting the Set ID to 99 indicates that the tape is not to be used by DIVArchive. This particularly applies to cleaning tapes installed in the library if they are reported to DIVArchive after a library audit (typical barcode is CLNXXXX for example).

This also applies to some installations where DIVArchive shares its libraries with other applications. Tapes in use by those applications should also have their Set ID set to 99 to prevent DIVArchive from using them.

The **Edit Multiple Rows** dialogue appears when multiple tapes are selected and the **Edit** button is selected. The Set ID will be updated on all tapes selected.

4.23 Media Remapping

The **Media Mapping** frame in the **Sets, Groups, & Media Mapping** tab allows you to put transformation rules in place for the specified groups on archive requests. The remapped destination media can be either a Disk Array, Tape Group or a Storage Plan. This would not typically be used during initial installation, but some time later in the archive's life cycle.

The transformation rules allow transparent redirection of objects from one media type to another without needing to alter the archive initiator (*such as a MAM or automation system*). Some examples are migration of an existing group to a new drive/tape generation, or migration from tape to disk.

Figure 111: Media Remapping

The screenshot shows the 'Media Mapping' dialog box. At the top, there is a table with columns: Id, Name, From, Map to Media, and Map to Storage Plan. The first row contains the values: 11, hant, array_default, and SP_DEFAULT. Below the table is a 'New row in Media Mapping' section with the following fields:

- Id:** 23
- Name:** (empty text box)
- From:** array_axf
- Map to Media:** (dropdown menu)
- Map to Storage Plan:** (dropdown menu)

Callouts provide the following explanations:

- Symbolic name for the mapping rule.** (points to the Name field)
- This mapping rule will be applied when an archive request is received matching this Destination Media name.** (points to the From field)
- Storage Plan to remap the request to. If this field is left blank, it will default to SP_DEFAULT. This may override the filters of that Storage Plan if they are based on the same Source Media.** (points to the Map to Storage Plan field)
- Destination Media to remap to. This will override the Destination Media specified.** (points to the Map to Media field)

Buttons for 'OK' and 'Cancel' are located at the bottom of the dialog.

The following events appear in the request details when an object's media is remapped to another media, or a storage plan, or both:

- Media Name Translation has changed the Destination Media to **media**.
- Media Name Translation has changed the Destination Media to **storageplan**.
- Media Name Translation has changed the Destination Media to **media & storageplan**.

5 DIVArchive Actor Configuration

The DIVArchive Actor on Windows platforms runs as a Windows Service and is launched automatically with Windows. The DIVArchive Actor runs as a standalone server application; the DIVArchive Manager connects to each Actor as a client application. The Actor is installed in the `DIVA_HOME\Program\Actor\bin` folder with configuration files located separately in the `DIVA_HOME\Program\conf\Actor` folder.

At the system level, the location and capabilities of each DIVArchive Actor are defined in the Configuration Utility.

5.1 File Path Limitations

DIVArchive cannot open or create files on a Windows filesystem if their absolute path exceeds 260 characters. A local path is structured in the following order and terminated with a `NUL` character:

```
Drive_Letter:\Component_Name\Component_Name\filename.extension
```

Examples:

- Generic
`D:\Some_256-Character_Path_String<NUL>`
- Actor Executable
`D:\diva\71\Program\Actor\bin\DIVArchive.exe`
- Manager Configuration
`D:\diva\71\Program\conf\manager\manager.conf`

The `<NUL>` character used in the examples represents the invisible terminating null character for the current system code page and need not be typed in. The `<` and `>` characters are used in the examples for clarity only and cannot be part of a valid path string.

5.2 Configuration Overview

The Actor configuration settings have been moved from the actor configuration and partial file restore configuration files to the Configuration Utility with the exception of the Service Name and Port. These settings are located under Actor Advanced and Partial Restore Settings tabs of the Actor Panel of the Systems Tab. Some settings are only available In Engineering Mode.

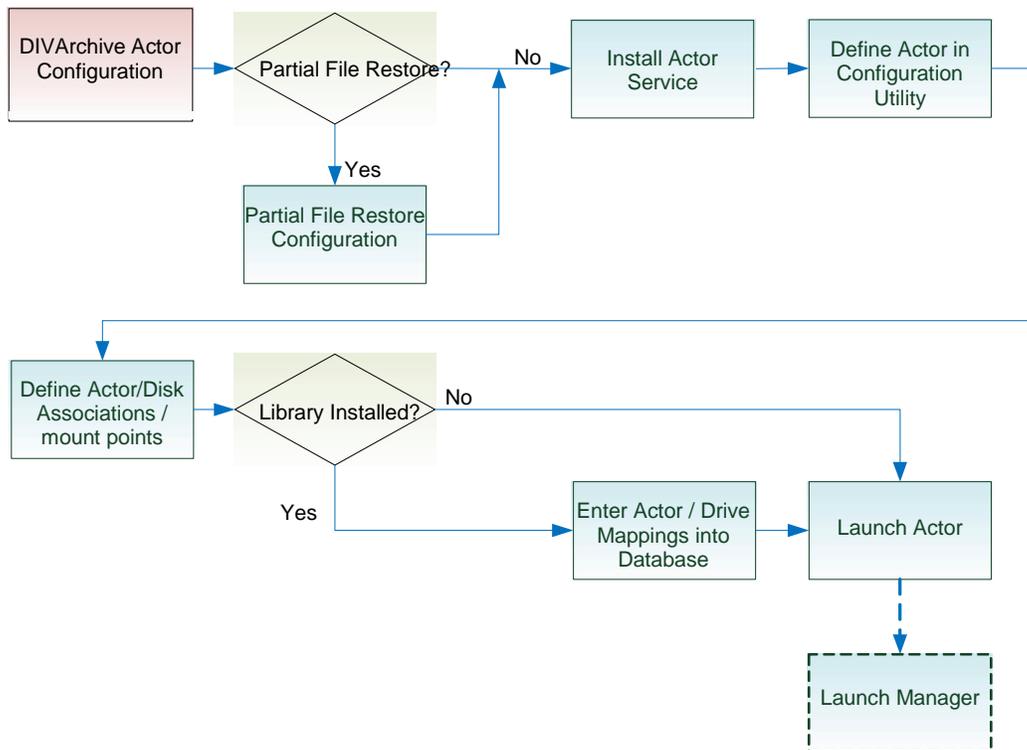
You must notify the actors of any changes to the configuration by clicking on **Notification**, **Notify Actors** while connected to the Manager. The actors must be running and connected to the Manager to receive the notifications. A list of what can be applied dynamically to an online system is described in **Error! Reference source not found.**

Figure 112: Notifying Actor of Configuration Changes



This is the workflow for a new DIVArchive Actor installation:

Figure 113: Actor Configuration Overview



5.3 Local DIVArchive Actor Configuration (*actor.conf*)

Starting with DIVArchive 7.3 the Actor configuration parameters have been moved from the Actor configuration and Partial File Restore configuration files to the Configuration Utility with the exception of the Service Name and Port.

Remove the .ini extension from the *actor.conf.ini* file and edit the file to insert the Actor Service Name and Port number as described in the next table.

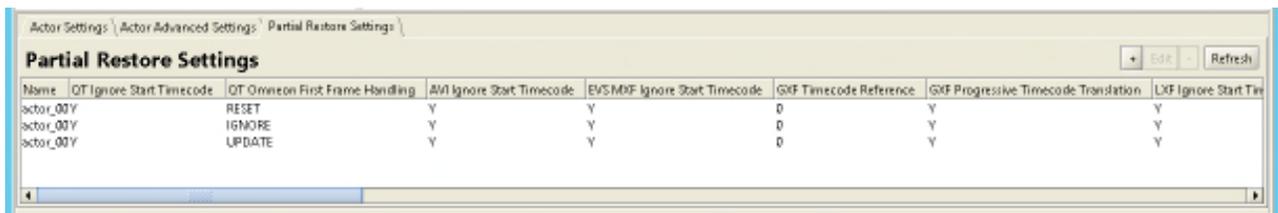
Table 27: Local DIVArchive Actor Configuration Options in *actor.conf*

DIVArchive Actor General Options	Parameter Type	Default
DIVAActor_PORT	TCP Port Number	9900
The TCP Port Number for the Actor to listen on for incoming requests. If running more than one Actor on the host the TCP Port Number should be unique for each Actor.		
SERVICE_NAME	Name	
The DIVAActor_SERVICE_NAME parameter specifies the name of the Actor and of the Windows Service during installation. This would be required if you need to install two or more Actors on a single Windows host (<i>since both cannot have the same Actor/Service Name</i>). If this parameter is not defined or commented out, the Service Name defaults to the Hostname of the Actor machine and will be DivaAct [HOSTNAME] .		

5.4 Partial File Restore Configuration

Starting with DIVArchive 7.3, the parameters in the *partial_restore.conf* file are located on the **Partial Restore Settings Tab** in the Configuration Utility's Actor Panel. These options provide additional parameters to the Actor for specific partial file restore formats.

Figure 114: Configuration Utility Partial Restore Parameters



Double-clicking the Actor Name on the Partial Restore Settings Tab will open the Actor Advanced Settings screen. Partial restore options are identified on the Partial Restore Settings Tab of the screen.

Figure 115: Actor Partial Restore Settings Tab

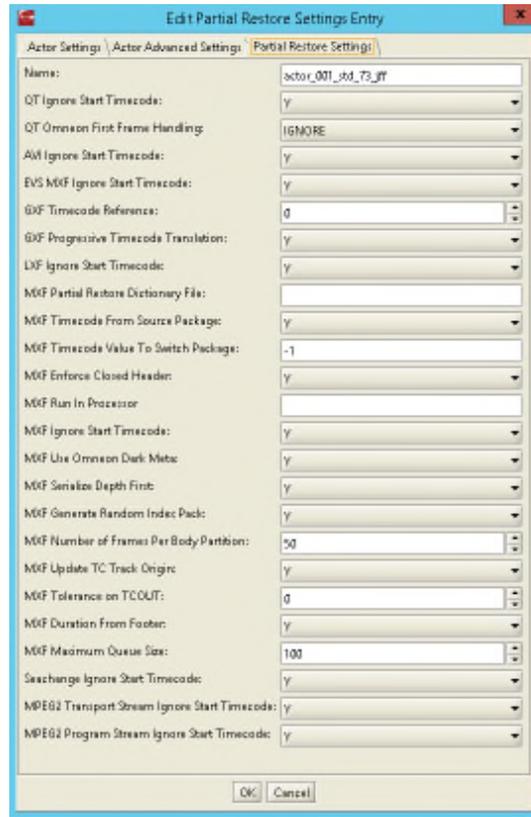


Table 28: Partial Restore Parameters

Parameter	Value / Type	Default
Name	String	
<p>This is the name of the Actor associated with these Partial File Restore options. This value is automatically filled in from the Actor Settings. If you modify the name here, or in the Actor Settings Screen, it will be modified in both places.</p>		
QT Ignore Start Timecode	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>If this setting is enabled, partial file restore will ignore the SOM value of the original clip and process TCIN and TCOUT as if it starts from 00:00:00:00. By default, this option is disabled.</p>		

Parameter	Value / Type	Default
QT Omneon First Frame Handling	IGNORE RESET UPDATE	RESET
<p>This setting identifies how the actor will handle the first frame of a QuickTime clip:</p> <ul style="list-style-type: none"> • IGNORE: Partial restore will ignore this field. The value found in the original clip will remain unchanged in the restored clip. • RESET: Partial restore will reset the value of this field to zero. This is the default behavior. • UPDATE: Partial restore will increment this value using the frame count from which the partially restored file begins. 		
AVI Ignore Start Timecode	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>If this setting is enabled, partial file restore will ignore the SOM value of the original clip and process <code>TCIN</code> and <code>TCOUT</code> as if it starts from 00:00:00:00. By default, this option is disabled.</p>		
EVS MXF Ignore Start Timecode	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>If this setting is enabled, partial file restore will ignore the SOM value of the original clip and process <code>TCIN</code> and <code>TCOUT</code> as if it starts from 00:00:00:00. By default, this option is disabled.</p>		
GXF Timecode Reference	Integer	1
<p>This setting specifies how the time code SOM reference is to be derived for a GXF partial file restore request. The options are defined by the following values:</p> <ul style="list-style-type: none"> • The objects start time codes are ignored. <code>TCIN</code> and <code>TCOUT</code> must be relative to 00:00:00:00. • SOM is derived from the first field number of the MAP packet (<i>default</i>). • SOM is derived from the time code at Mark In from the UMF packet. 		

Parameter	Value / Type	Default
GXF Progressive Timecode Translation	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>By default Partial File Restore is expecting τCIN and τCOUT to be in conformance with the frame rate of the archived clip. For example, if the frame rate of the clip is 29.97fps NTSC (or 25fps for PAL), the frame count of τCIN and τCOUT can be comprised between 0 and 29 (25 if it is PAL).</p> <p>HD formats now have progressive frame rates (23.976, 24, 29.97, 30, 59.94, 60). For automations, the actual frame rate of the clip can be unknown. When this parameter is set to Y (<i>enabled</i>), DIVArchive will consider that τCIN and τCOUT are PAL or NTSC timecodes and will translate these timecodes according to the actual frame rate of the archived clip.</p>		
LXF Ignore Start Timecode	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>If this setting is enabled, partial file restore will ignore the SOM value of the original clip and process τCIN and τCOUT as if it starts from 00:00:00:00. By default, this option is disabled.</p>		
MXF Partial Restore Dictionary File	Path and Filename	
<p>This parameter must point to the name and location of the MXF dictionary file. The dictionary is normally distributed with the DIVArchive Actor installation (e.g. <code>DIVA_HOME\Program\Actor\bin</code>). The dictionary file is named <code>mxf_file.bin</code> by default.</p> <p>Set this parameter to <code>DIVA_HOME\Program\Actor\bin\mxf_file.bin</code>.</p> <p>Where <code>DIVA_HOME</code> is the root path of your DIVArchive installation for the Actor (<i>typically</i> <code>c:\Diva</code> for Windows).</p>		
MXF Timecode From Source Package	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>If this parameter is set to Y (<i>enabled</i>) the time code track used to locate the in/out points will be the one from the source package. The default is N (<i>disabled</i>) - timecode will be sourced from the Material Package.</p>		

Parameter	Value / Type	Default
MXF Timecode Value To Switch Package	-1 (<i>no switch</i>) 0 (<i>switch</i>)	-1
<p>If the SOM value found in the MXF package specified by the parameter <i>MXF Timecode From Source Package</i> is equal to this value, the actor will automatically look for the SOM in the other MXF PACKAGE. The default value of -1 avoids switching from one package to the other.</p>		
MXF Enforce Closed Header	N (<i>disabled</i>) Y (<i>enabled</i>)	Y
<p>If this parameter is set to Y (<i>enabled</i>) the extraction will fail if the Metadata in the header is not closed. If not set, or set to another value, the Actor will try to find closed Metadata in the footer partition. The default value is Y.</p>		
MXF Run In Processor	Filename	
<p>If this parameter is defined, it must contain the name of the <code>RunInProcessor.DLL</code>. In this case, the run-in processor will be used to read and create run-ins. Example: <code>RUN_IN_PROCESSOR=RunInProcessor.dll</code>.</p>		
MXF Ignore Start Timecode	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>If this parameter is set to Y, MXF partial file restore will ignore all start time code values of the original clip and <code>TCIN</code> and <code>TCOUT</code> (<i>SOM and EOM</i>) will be processed as if the original clip starts from 00:00:00:00. This option overrides the <code>MXF TIMECODE FROM SOURCE PACKAGE</code> parameter.</p>		
MXF Use Omneon Dark Meta	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>Certain Omneon MXF clips have their start time code located in a Dark Metadata Set. By default the MXF partial file restore does not pay attention to this field. If you want the MXF partial file restore to take care of this field then set this parameter to Y.</p>		

Parameter	Value / Type	Default
MXF Serialize Depth First	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>If this parameter is set to Y (<i>enabled</i>) the MXF partial file restore serializes the Metadata Sets of the partial file restored clip using a depth first approach. This option is recommended when the destination is a QUANTEL ISA gateway. If it is set to N (<i>disabled</i>), the MXF partial file restore serializes the Metadata Sets with no ordering; this is the default behavior.</p>		
MXF Generate Random Index Pack	N (<i>disabled</i>) Y (<i>enabled</i>)	Y
<p>Random Index Pack (<i>RIP</i>) is a small structure located at the end of an MXF file. It is optional and when present contains file offset information for each partition in the file. RIP can be disabled (<i>by setting this parameter to N</i>) for those servers not compatible (<i>such as SONY XDCAM</i>). By default, this setting is enabled.</p>		
MXF Number of Frames Per Body Partition	Integer (50~250 only)	250
<p>This parameter defines the number of frames per partition in the output file. Only values between 50 and 250 are valid. If a value greater than 250 is entered the MXF partial file restore will use 250, and if it is lower than 50, it will use 50. Note: This parameter is rounded automatically by the Actor in order to align body partitions on GOP boundaries.</p>		
MXF Update TC Track Origin	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>When the video essence is MPEG2 LGOP, partial file restore will use the origin field of each track in order to be frame accurate. The origin is used to specify GOP precharge frames. Your video server may use a different implementation or interpretation of this field. If this parameter is set to Y (<i>enabled</i>), the Origin field is modified in all tracks. If this parameter is set to N (<i>disabled</i>), the Origin field is modified in all tracks <i>except</i> the timecode track.</p>		

Parameter	Value / Type	Default
MXF Tolerance on TCOU	Integer (0 - 250 only)	0
<p>This parameter can be set to indicate a tolerance on the <code>TCOU</code> supplied to a partial file restore request. By default this tolerance is 0, but it can be set to a number of frames and if the supplied <code>TCOU</code> is beyond the end of the clip but not too far out (<i>in the tolerance</i>), then DIVArchive will do the partial file restore until the end of the clip instead of reporting <i>Invalid TCOU</i>. The maximum value allowed for this parameter is 250 frames.</p>		
MXF Duration From Footer	N (disabled) Y (enabled)	Y
<p>When the duration of the input clip is -1 in the header partition, the MXF partial file restore will load the footer partition in order to get the correct value. Some older clips may not have a correct RIP at the end of the file and the footer partition may not be accessible.</p> <p>If the MXF Duration From Footer is set to N, the MXF partial file restore will not load the footer partition and will perform a <i>blind</i> partial file restore assuming that <code>TCIN</code> and <code>TCOU</code> are valid.</p>		
MXF Maximum Queue Size	Integer (0 - 200 only)	200
<p>Max size in MB that the extractor can queue before throwing an error (<i>to avoid running out of memory</i>).</p>		
Seachange Ignore Start Timecode	N (disabled) Y (enabled)	N
<p>If the parameter <code>SEA IGNORE START TIMECODE</code> is set to Y, SeaChange partial file restore ignores the start time code value of the original clip and processes <code>TCIN</code> and <code>TCOU</code> as if it starts from 00:00:00:00. In the case of MXF, the configuration of the MXF parser is also required. The parameter <code>MXF IGNORE START TIMECODE</code> is ignored in this workflow because this is a Seachange clip. By default this option is disabled.</p>		

Parameter	Value / Type	Default
MPEG2 Transport Stream Ignore Start Timecode	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>If this parameter is set to Y, the MPEG2 transport stream partial file restore will ignore the start time code value of the original clip and process <code>TCIN</code> and <code>TCOUT</code> as if it starts from 00:00:00:00. By default this option is disabled.</p>		
MPEG2 Program Stream Ignore Start Timecode	N (<i>disabled</i>) Y (<i>enabled</i>)	N
<p>If this parameter is set to Y (<i>enabled</i>), MPEG2 transport stream partial file restore will ignore the start timecode value of the original clip and process <code>TCIN</code> and <code>TCOUT</code> as if it starts from 00:00:00:00.</p>		

5.5 Installing/Removing DIVArchive Actor Service(s)

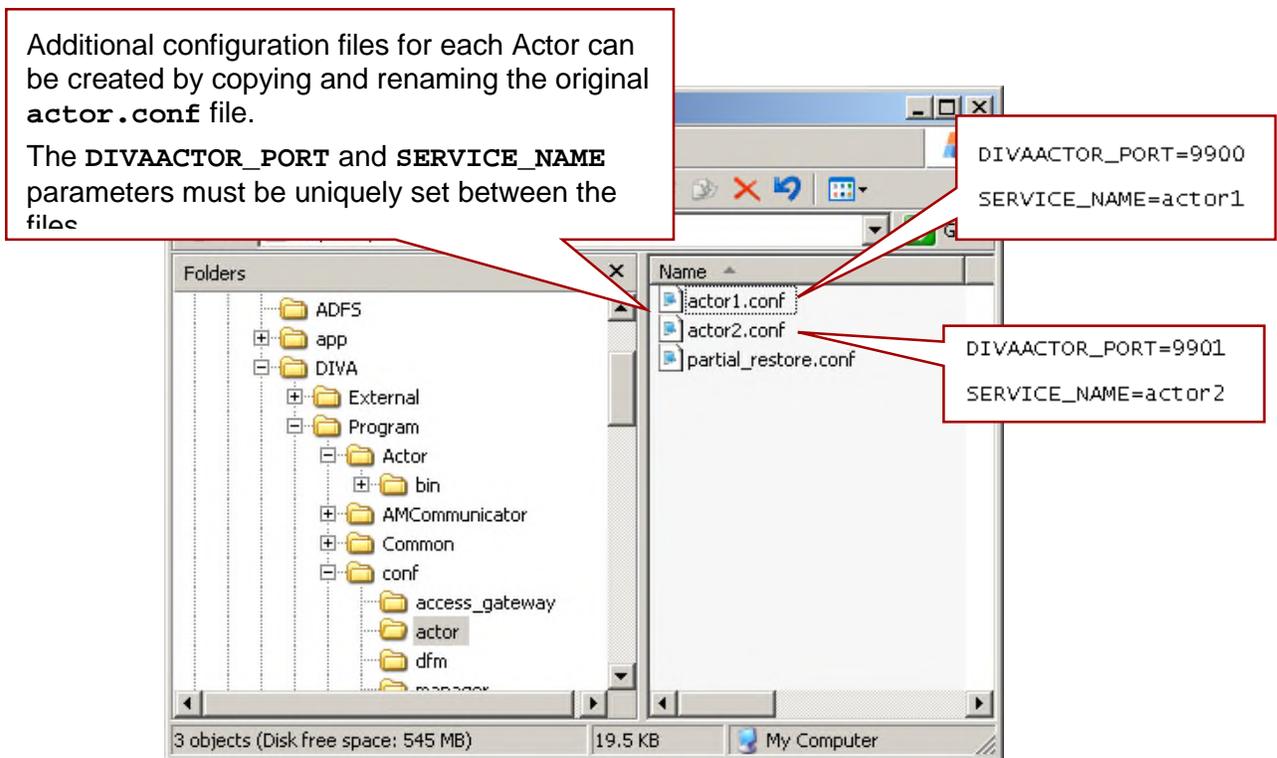
The `actorservice.exe` executable in the Actor bin directory is used to install (or *uninstall*) the DIVArchive Actor as a Windows Service from a Windows command prompt.

`actorservice -i` Installs the Actor Service using the `service_name` parameter defined in `actor.conf`. If this parameter is undefined, then the service is installed as `DIVArchive Actor - hostname`.

`actorservice -u` Removes the Actor Service using the `service_name` parameter defined in `actor.conf`. If this parameter is undefined, then the service that has to be removed is `DIVArchive Actor - hostname`.

By default, the Actor Service uses the `actor.conf` file located in `DIVA_HOME\Program\conf\actor` to define the Service Name. If you are installing multiple Actors on a single host, additional Actor Configuration Files must be created and specified to the service in order to create unique instances for each Actor.

Figure 116: Installing/Removing DIVArchive Actor Service(s) in Windows



When installing additional Actor Services on the same host the path to each Actor's configuration file needs to be specified for each instance. This is done by adding the `-conf` (or `-f`) command switches when installing the service:

```
actorservice -i -conf ..\..\conf\actor\actor1.conf
```

Installs the Actor Service defined by the `service_name` parameter in `actor1.conf`.

```
actorservice -i -conf ..\..\conf\actor\actor2.conf
```

Installs the Actor Service defined by the `service_name` parameter in `actor2.conf`.

If one or more Actor Services need to be uninstalled, the configuration file path needs to be specified:

```
actorservice -u -conf ..\..\conf\actor\actor1.conf
```

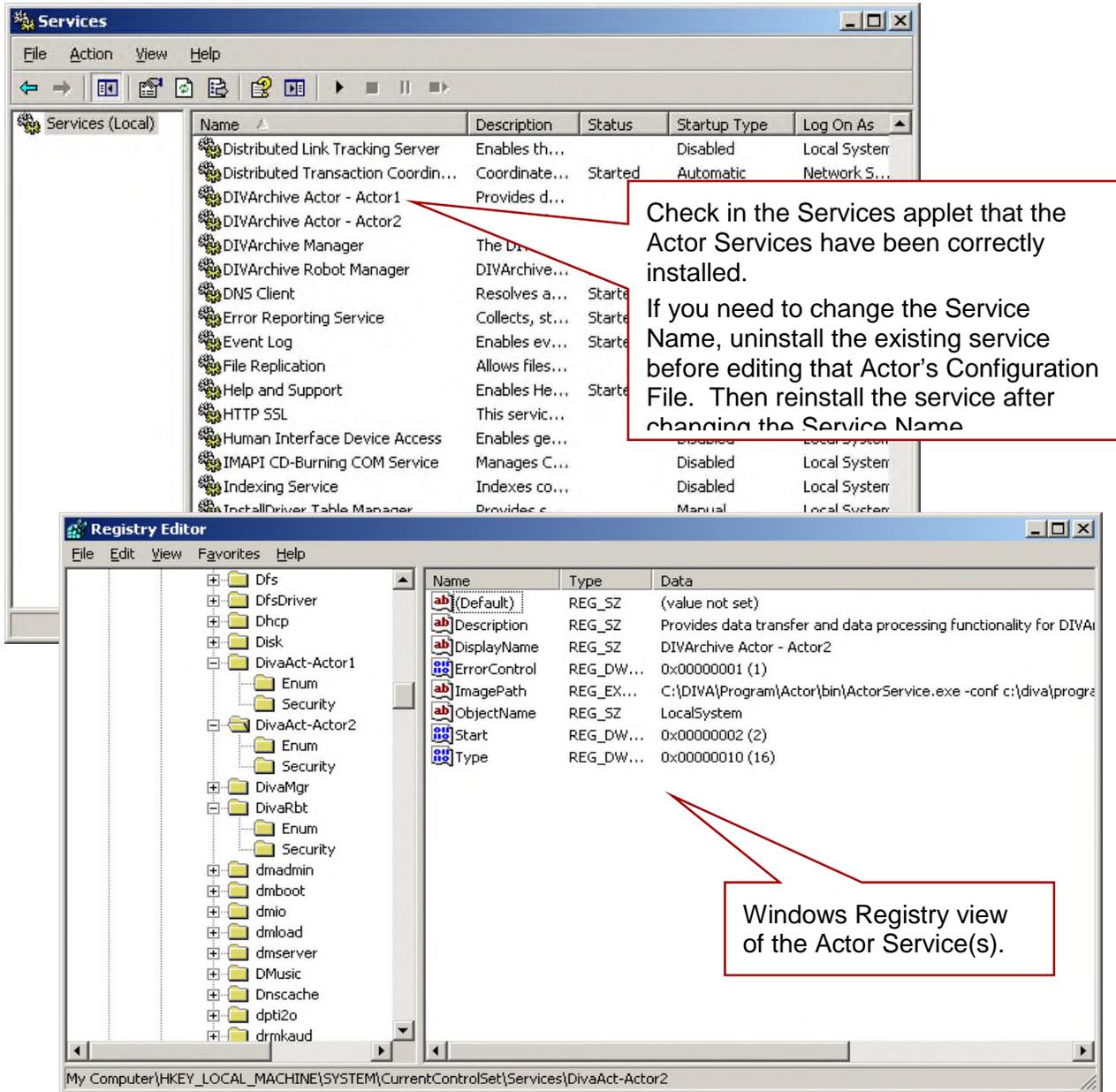
Removes the Actor Service defined by the `service_name` parameter in `actor1.conf`.

```
actorservice -u -conf ..\..\conf\actor\actor2.conf
```

Removes the Actor Service defined by the `service_name` parameter in `actor2.conf`.

After installing the service(s), check the Windows Services applet to verify that each Actor Service was correctly installed as shown in the figure on the next page.

Figure 117: Windows Services Applet and Registry Editor



5.5.1 Actor Service Management Functions

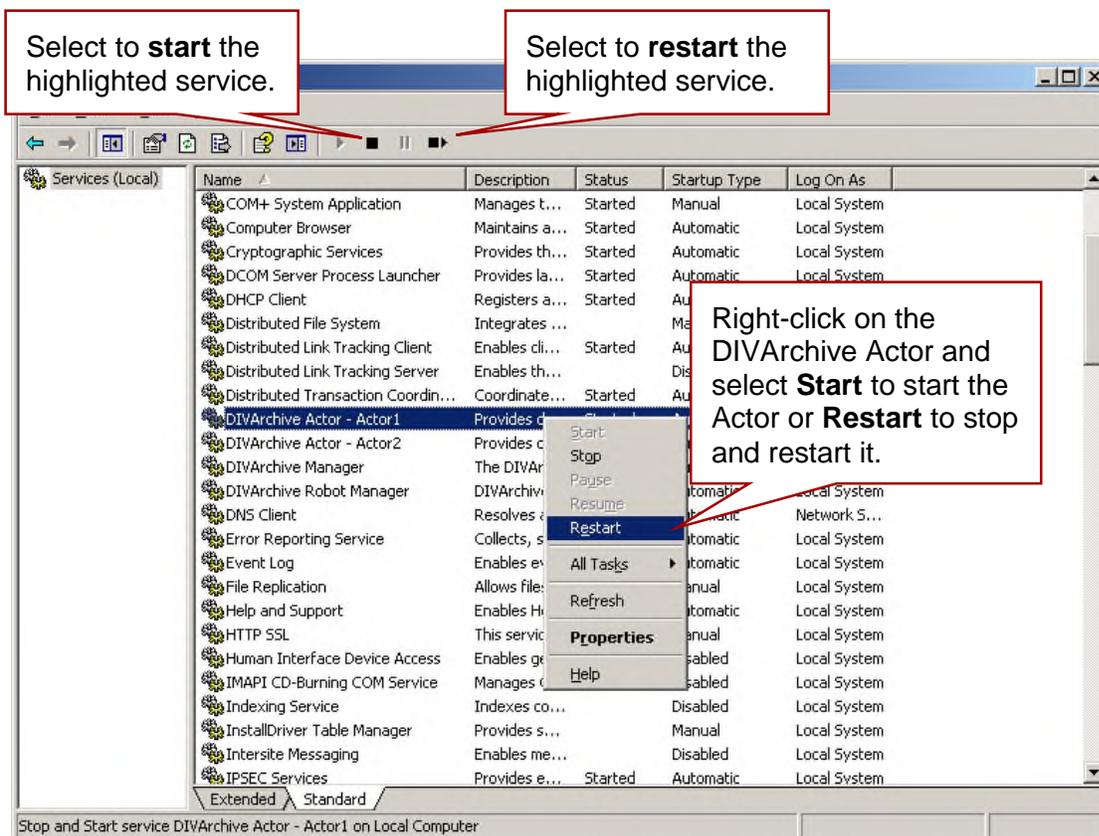
The following command options are also available for the Actor Service:

- actorservice debug** Starts the DIVArchive Actor Service in console mode.
- actorservice version (or -v)** Displays the DIVArchive Actor Service software version info.
- actorservice help** Displays all command line options.

5.6 Launching the DIVArchive Actor(s)

Windows DIVArchive Actors no longer start automatically with Windows. The Actor Services can be managed through the Windows Services applet.

Figure 118: Launching the DIVArchive Actor(s) in Windows Service Manager



An alternative way to restart an Actor is to execute the following from a command window (*Start Menu, Run, cmd.exe*).

```
net stop "DIVArchive Actor"  
net start "DIVArchive Actor"
```

If the `SERVICE_NAME` has been specified in the `actor.conf` file (see 5.3) then:

```
net stop "DIVArchive Actor -SERVICE_NAME"  
net start "DIVArchive Actor -SERVICE_NAME"
```

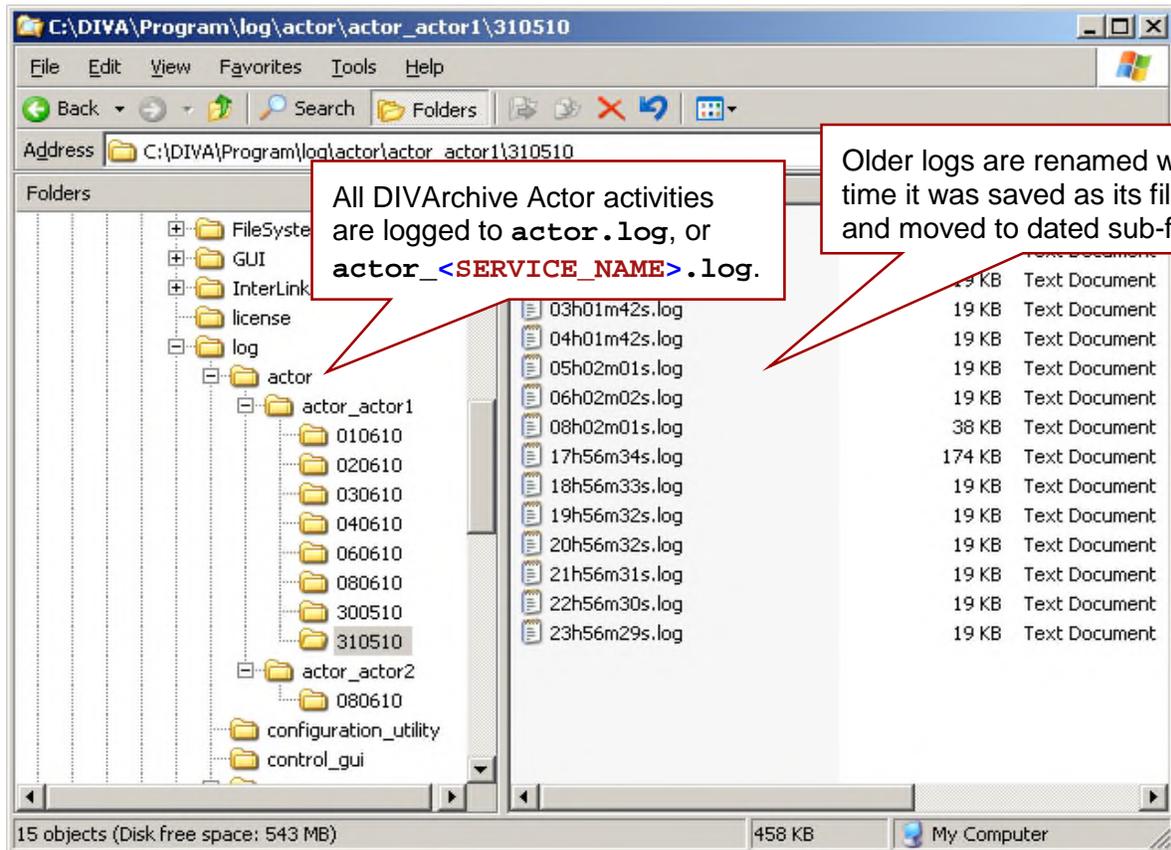
You should create a batch file using the command line options detailed above and put a shortcut to start the services on your desktop.

Note: The quotation marks in the above commands *must* be used when specifying a Windows service with spaces in the name.

5.7 Logging

During operation, each DIVArchive Actor logs all of its operations, and stores them under the `DIVA_HOME\Program\log\actor` folder. These files are useful in diagnosing transfer errors with either drives or servers and particularly for debugging the configuration for a newly added Source or Destination. You may be asked for these logs when contacting Oracle Technical Support for assistance.

Figure 119: DIVArchive Actor Logs Window



Note: Each DIVArchive Actor also provides additional logging functions for some specific server protocols (such as the *Quantel QCP interface, FTP Servers and Partial File Restore*). These logs are unique for each server type and are enabled by default. They provide detailed logging information from that protocol to the standard Actor log file.

5.8 DIVArchive Actor Definitions (Configuration Utility)

The Actor configuration settings have been moved from the actor configuration and partial file restore configuration files to the Configuration Utility with the exception of the Service Name and Port. These settings are located under Actor Advanced and Partial Restore Settings tabs of the Actor Panel of the Systems Tab. Some settings are only available In Engineering Mode.

5.8.1 DIVArchive Actor Declaration

To be utilized by DIVArchive, each DIVArchive Actor must be declared in the DIVArchive Database. This declaration is performed in the **Actors** window in the **System** tab of the Configuration Utility. The new Actors window has three tabs:

- Actor Settings includes general actor definition settings such as name, IP address, port, production system, and so on.
- Actor Advanced Settings includes advanced settings such as read and write block sizes, tape unit timeout, Quantel, QuickTime and FTP settings.
- Partial Restore Settings includes partial file restore settings previously in the partial file restore configuration file.

Figure 120: DIVArchive Actor Declaration

The screenshot shows the 'Actor Settings' window with a table of actor definitions. Three callout boxes provide instructions for the buttons at the top right of the window:

- Existing Actor Definitions**: Points to the table of actor settings.
- + Add**: Adds a new DIVArchive Actor Definition.
- Edit**: Edits the selected Actors parameters.
- Remove**: Removes the selected Actor. Note that this removal can only be achieved if the Actor is **NOT** referenced by any other DIVArchive configuration, such as an Actor/Tape or Actor/Disk binding.
- Refresh**: Select **Refresh** to refresh these details from the DIVArchive Database.

Name	IP Address	Port	Prod. System	Site	Drive Ops	Server Ops	Disk Ops	Verify Tape	Direct Restore	Cache Restore	Copy To Group	Associative Copy	Repack	Delete	Direct Archive	C
actor_001_std_73_lau	127.0.0.1	9901	ps_001	local	10	10	10	Y	Y	Y	Y	Y	Y	Y	Y	Y
actor_002_std_73_lau	127.0.0.1	9902	ps_001	local	10	10	10	Y	Y	Y	Y	Y	Y	Y	Y	Y
actor_003_std_73_lau	127.0.0.1	9903	ps_001	local	10	10	10	Y	Y	Y	Y	Y	Y	Y	Y	Y

Actor and Partial File Restore settings are configured and edited on the Actor Settings Entry screen. Click the **Plus Sign** on the top right of the Actor Settings Panel to create and configure a new Actor, or double-click the actor you want to edit to access the settings screen.

Figure 121: Actor Settings Entry Screen

Table 29: Maximum Operation Option Definitions

Parameter	Description
Name	This is the name of the Actor associated with these Partial File Restore options. This value is automatically filled in from the Actor Settings. If you modify the name here, or in the Actor Settings Screen, it will be modified in both places.
IP Address	This is the IP address of the Actor.
Port	This is the port number the Actor listens on for commands.
Prod. System	This parameter identifies the production system where the actor is in use.
Site	This parameter identifies the physical location of the production system.
Max Drive Operations	This is the maximum number of simultaneous requests dealing with drives that this Actor can perform. This parameter can be used to distribute requests and bandwidth amongst all Actors.

Parameter	Description
Max Server Operations	This is the maximum number of simultaneous requests dealing with servers from the Sources and Destinations configuration that this Actor can perform. This parameter can be used to distribute requests and bandwidth amongst all Actors.
Max Disk Operations	This is the maximum number of simultaneous transfers to disks (<i>both read and write</i>) that this Actor can perform. This parameter can be used to distribute requests and bandwidth amongst all Actors.
Verify Tape	This parameter defines whether or not tapes are verified.
Direct Restore	Defines whether this Actor can be used for direct restores to a Source or Destination.
Cache Restore	The Actor is permitted to perform cache restores to a Source or Destination. Disable this option if this Actor has no local cache storage for the temporary storage of the DIVArchive Object during a transfer.
Copy To Group	Defines whether this Actor can be used for Copy to Group requests. This option could be used to isolate specific Actors involved in critical operations from mass Copy to Group requests, such as those from the DIVArchive SPM option.
Associative Copy	Defines whether this Actor can be used for Associative Copy requests.
Repack	Defines whether this Actor can be used for Tape Repack requests. If the Actor has no local cache for temporary storage during the repack operation, set this to N . Since tape repacking is a lengthy operation it could also be used to dedicate an Actor solely to repack requests by disabling the other options (<i>except Delete</i>) and disabling Repack on the other Actors.
Delete	Defines whether this Actor can be used for requests that involve deleting DIVArchive Objects from a disk. This option could be used to isolate an Actor from mass deletion requests (<i>such as those issued from the Storage Plan Manager option</i>).
Direct Archive	Defines whether this Actor can be used for Direct Archive requests.

Parameter	Description
Cache Archive	Defines whether this Actor can be used for Cache Archive requests. Disable this option if this Actor has no local cache storage for the temporary storage of the DIVArchive Object during a transfer.
First Utilization Date	This is the date the actor was first put into use.

5.8.1.1 Actor Advanced Settings Panel

Advanced Actor parameters are displayed on the **Actor Advanced Setting Tab** in the **Actors Panel** of the Configuration Utility. Entries are configured and edited on the Actor Settings Entry screen's **Actor Advanced Settings Tab**. To configure or edit advanced actor parameters, double-click the actor you want to edit to access the settings screen.

Figure 122: Actor Advanced Settings Panel

Name	Tape Test Unit Ready Timeout (s)	Profile Read Block Size (B)	Profile Write Block Size (B)	Quantel Rename Clips	QT Self-contained Threshold (MB)	Disk FTP Passive Mode	Disk FTP Block Size
actor_001_std_73_lou	60	1500	1500	Y	10	Y	32
actor_002_std_73_lou	60	1500	1500	Y	10	Y	32
actor_003_std_73_lou	60	1500	1500	Y	10	Y	32

Figure 123: Actor Advanced Settings Tab

Edit Actor Advanced Settings Entry

Actor Settings \ Actor Advanced Settings \ Partial Restore Settings \

Name: actor_001_std_73_lou

Tape Test Unit Ready Timeout (s): 60

Profile Read Block Size (B): 1500

Profile Write Block Size (B): 1500

Quantel Rename Clips: Y

QT Self-contained Threshold (MB): 10

Disk FTP Passive Mode: Y

Disk FTP Block Size (KB): 32

Disk FTP Socket Window Size (B): 65536

OK Cancel

Table 30: Actor Advanced Parameters

Parameter	Description
Name	This is the name of the Actor associated with these Partial File Restore options. This value is automatically filled in from the Actor Settings. If you modify the name here, or in the Actor Settings Screen, it will be modified in both places.
Tape Test Unit Ready Timeout (s)	Time in seconds to allow for a drive to become ready once a tape is mounted; after which the drive is considered to be not responding.
Profile Read Block Size (B)	<p>FTP block size used for transfer on profile video servers when reading. The default value (1500) has been determined to be the best blocksize with GVG profile servers; however this value may be different when using other servers.</p> <p>Possible values are between 1500 and 262144 bytes.</p>
Profile Write Block Size (B)	<p>FTP block size used for transfer on profile video servers when writing. The default value (32768) has been determined to be the best blocksize with GVG profile servers; however this value may be different when using other servers.</p> <p>Possible values are between 1500 and 262144 bytes.</p>
Quantel Rename Clips	<p>Automatically rename clips when restoring them to Quantel.</p> <ul style="list-style-type: none"> • N – this feature is disabled (<i>default</i>). • Y – files are renamed using the objectname truncated (<i>first part before the comma (,): Omnibus renaming</i>)
QT Self-contained Threshold (MB)	When doing a QuickTime Partial File Restore, the actor has to determine if a clip is self-contained or not based on the size of the input file. This parameter is a limit (<i>in MB</i>) beyond which the actor will consider the clip to be self-contained. The unique objective of this parameter is to prevent the actor from loading a large self-contained clip into memory. Values range from 10MB through 100MB.

Parameter	Description
Disk FTP Passive Mode	<p>FTP data connections are created in active mode by default. This means that the DivaFTP client connects from a random unprivileged port (<i>greater than 1023</i>). Then it immediately starts listening to the port and sends a <code>PORT</code> command to the FTP server.</p> <p>Setting this parameter to Y causes the data connections to be created in passive mode rather than active mode. This causes the DivaFTP client to send a <code>PASV</code> command to the FTP server and the socket is created by the server (<i>not the client</i>).</p>
Disk FTP Block Size (KB)	<p>This parameter defines how much data the actor retries to send/receive with a single system call during FTP transfers.</p> <p>Example:</p> <p>If the internal buffer size of the actor is set to 2MB, and this parameter is set to 32768 bytes, 64 system calls are required to write a single buffer to a data socket.</p>
Disk FTP Socket Window Size (B)	<p>This parameter adjusts the normal buffer size allocated for output and input buffers. This parameter is internally used to set the send and receive buffers for FTP managed disk types.</p>

5.9 Actor to Disk Definitions

After the Actor Definitions have been configured, the logical connections (*or mount points*) of the physical disks previously defined during the initial DIVArchive configuration can be defined.

Note: If the same resource on a physical disk is to be shared between multiple Actors (*and file sharing software has been installed*) we recommend that the drive letter or volume of the disk connection configured in each Actor host be identical for simplicity.

Figure 124: Actor to Disk Definitions

The screenshot shows a web-based interface titled "Actor-Disk Connections". It features a table with columns for Disk Id, Disk, Actor, Interface, Mount point, Throughput, Access, and Usage. The table contains five rows of data. Above the table, there are three callout boxes: one pointing to the table header, one pointing to the table body, and one pointing to the "+ Edit - Refresh" buttons.

Existing Actor-Disk Connection Definitions.

The Actors retrieve these definitions when the DIVArchive Manager first connects to each of them. Any modifications performed here require the relevant Actor to be restarted.

Adds, removes, or edits an Actor-Disk Connection Definition.

Disk Id	Disk	Actor	Interface	Mount point	Throughput	Access	Usage
1	disk_axf	actor1_70	local	D:\diva\70\simulator\disks\disk1	100,000	Read/Write	CACHE_AND_STORAGE_AND_NEARLINE
3	disk_axf_vw	actor1_70	local	D:\diva\70\simulator\disks\disk1	100,000	Read/Write	CACHE_AND_STORAGE
4	disk_leg	actor1_70	local	D:\diva\70\simulator\disks\disk2	100,000	Read/Write	CACHE_AND_STORAGE_AND_NEARLINE
6	disk_leg_vw	actor1_70	local	D:\diva\70\simulator\disks\disk2	100,000	Read/Write	CACHE_AND_STORAGE
15	disk_axf_2	actor2_70	remote	C:\diva\70\simulator\disks\disk1	100,000	Read/Write	CACHE_AND_STORAGE_AND_NEARLINE

When a different Actor is selected, the drives available for configuration are displayed. If all drives have already been configured for the selected Actor, the Drives pull-down will not be available and will indicate that there are no drives available for the selected Actor.

Figure 125: Actor-Disk Add New Connections Window

The screenshot shows the 'Add New Connection in Actor-Disk' window. The fields and callouts are as follows:

- Actor:** A dropdown menu showing 'disk_axt'. Callout: "Select a physical disk in the drop-down menu for this Actor association. Only entries previously defined in the **Disks** frame will be displayed. Multiple disks may be selected using the checkbox next to each disk name."
- Actor:** A checkbox labeled 'actor2_70'. Callout: "Select the Actor for this disk association. Only Actors declared in the **Actors** frame of the **System** tab will be listed. The selected disk must be directly accessible by this Actor."
- Interface:** A text field containing 'local'. Callout: "The Mount Point is used in conjunction with the *Interface* selection"
- Mount Point:** An empty text field.
- Max. Throughput, MB/s:** A text field containing '100000'. Callout: "This allows bandwidth throttling of the transfers performed by the Actor. Typically used to load balance transfers with other Actors or non-DIVArchive applications. When DIVArchive has multiple disks to choose from for object storage, this parameter is the first criteria for disk allocation (*i.e. the disk with the highest throughput will be used first*). The second criterion is the percentage of used capacity of each disk considered."
- Access:** A dropdown menu showing 'Read/Write'. Callout: "This defines this Actor's read/write access to the associated logical disk. This allows further granularity in load balancing with other Actors."
- Used for:** A dropdown menu showing 'CACHE_AND_STORAGE_AND_NEARLINE'. Callout: "This defines how the associated disk is to be used by this Actor.
 - Cache only** DIVArchive will only use this disk for caching operations.
 - Storage only** DIVArchive will only use this disk for object storage.
 - Cache and Storage** DIVArchive will use this disk for both cache and object storage.
 - Storage and Nearline** DIVArchive will use this disk for both object and Nearline storage.
 - Cache and Storage and Nearline** DIVArchive

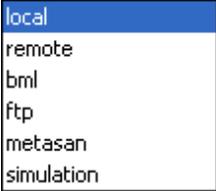
Note: Multiple selections are available in Add Mode, but not in Edit Mode. Nearline storage is storage used for Disk Instances that are created during a Restore or N-Restore with a Nearline QOS when no other Disk Instances are available.

5.10 Actor to Disk Interfaces and Mount Points

The Disk Interface method and the corresponding Mount Point in an Actor-Disk connection are determined by how the drive is logically connected and represented to the host operating system of the Actor.

Figure 126: Actor-Disk Interfaces and Mount Points

local	Unbuffered (direct) I/O will be used with the disk.
remote	Buffered I/O will be used with the disk.
bml	For Seachange Broadcast Media Libraries only.
ftp	Disk storage access via FTP protocol.
metasan	Disk is part of a Metasan installation.
simulation	For DIVArchive simulator platforms.



5.10.1 Local

This option specifies that **unbuffered I/O** will be used with the disk to maximize transfer performance.

Disks that use this option can reside within the Actor host itself (*such as disks to be used for cache purposes*), disks connected to the host via either SCSI or Fiber Channel HBA's (*such as those in a SAN*), or those specified with a **Universal Naming Convention (UNC)** Mount Point. Some network drives may actually suffer with this type of I/O; in this instance, use the **Remote** option instead.

Note: Windows based Actors do not support network drives mapped to a Windows drive letter (*this is a Microsoft security restriction*). Networked disks in Windows must use the Remote option instead.

The Mount Point is the drive letter or volume of the drive as it appears to the host operating system plus any additional directory path.

5.10.2 Remote

This option specifies that **buffered I/O** will be used with the disk, and allows access to disks hosted by another computer via the **Common Internet File System (CIFS)** protocol. This option **must** be used for networked disks with the Windows Actor Service.

The mount point for a CIFS connection is a UNC path, such as `cifs://192.168.56.26\shared` Or `cifs://user:password@//192.168.56.26\shared`.

Appropriate permissions for any CIFS based disk must be enabled for the Actor to access the network drive or the disk will be set **Offline**.

5.10.3 BML

This option allows the Actor to use a SeaChange BML (*non-Infiniband Media Libraries*) as disk storage. For regular disks, DIVArchive stores objects under a number of subdirectories. The BML however uses a flat file system (*i.e. no directory structure*). DIVArchive automatically incorporates a directory structure into the filename when it is archived to the BML and removes this addition from the filename as it is restored.

The mount point for the BML option is `bml://IP_ADDRESS`.

Example: `bml://10.201.10.124`

5.10.4 FTP

This option allows DIVArchive to use FTP Servers as disk storage, using the FTP protocol. The mount point must be in the format:

```
ftp://login:password@<host>/<rootdir>
```

Refer to the Managed Disk Configuration document for more information.

5.10.5 MetaSAN

This option should be selected when the volume is managed by MetaSAN. By default, DIVArchive Actors pre-allocate storage on disks to prevent disk fragmentation. MetaSAN implements its own mechanisms in this regard and selecting this option will disable pre-allocation when dealing with this volume.

5.10.6 Simulation

This option is used when setting up a DIVArchive Simulator (*refer to the DIVArchive Simulator Operations Guide for details – available to OPN partners only*).

The mount point must be a real pathname to a directory on a local disk. When used to store objects, only the file size is recorded to the disk (*i.e. no content is actually saved*). A simulated disk cannot be used as cache for a repack request.

5.11 Actor to Drive Configuration

Apart from the Tape Drive Control Configuration for the Robot Manager, the data transfer component of the drives must also be configured for use with the Actors. This involves the logical configuration of each drive in the Actor/Drive Configuration in the database.

Figure 127: Actor-Drive Configuration

Associated Actor of the drive mapping.

Drive Number and associated library where the drive is located.

Adds, removes, or edits an Actor/Drive binding.

Actor	Drive
actor_001_std_73_lou	0 (ACS=0, LSM=0)
actor_001_std_73_lou	1 (ACS=0, LSM=0)
actor_001_std_73_lou	2 (ACS=0, LSM=0)
actor_001_std_73_lou	3 (ACS=0, LSM=0)
actor_001_std_73_lou	4 (ACS=0, LSM=0)

If a drive is connected to multiple Actors via a SAN, the Actor/Drive mapping must be repeated for each Actor that will access this drive.

In combination with the Drive Operations Settings, the Actor Capability Settings can be used to dedicate a drive to a particular set of Actors for specific operations, such as tape repacking.

Figure 128: Add New Actor-Drive Connections Window Showing Available Connections

Actor: actor b2 70 1

Drives: 0 (ACS=0, LSM=0)

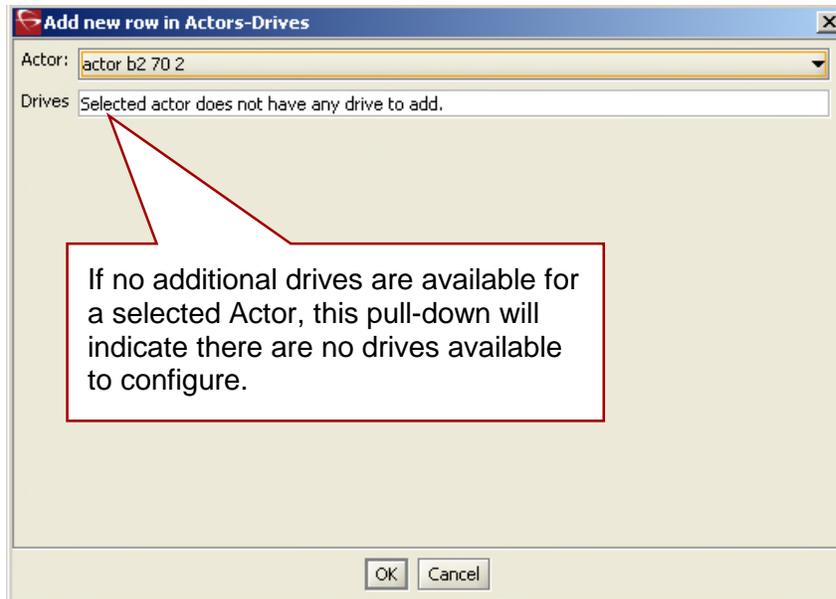
Selects the logical drive in the relevant library for this mapping. Only drives defined in the **Drives** frame of the **Drives** tab will be listed in the pull-down menu. Select multiple drives using the checkboxes.

To select the Actor that the drive is connected to use the pull-down menu. Only Actors already defined in the **Actors** frame of the **System** tab will be listed in the pull-down menu.

Note: Multiple selections are available in Add Mode, but not in Edit Mode.

When a different Actor is selected, the drives available for configuration are displayed. If all drives have already been configured for the selected Actor, the Drives pull-down will not be available and will indicate that there are no drives available for the selected Actor.

Figure 129: Add New Actor-Drive Connections Window Showing No Additional Connections Available for This Actor



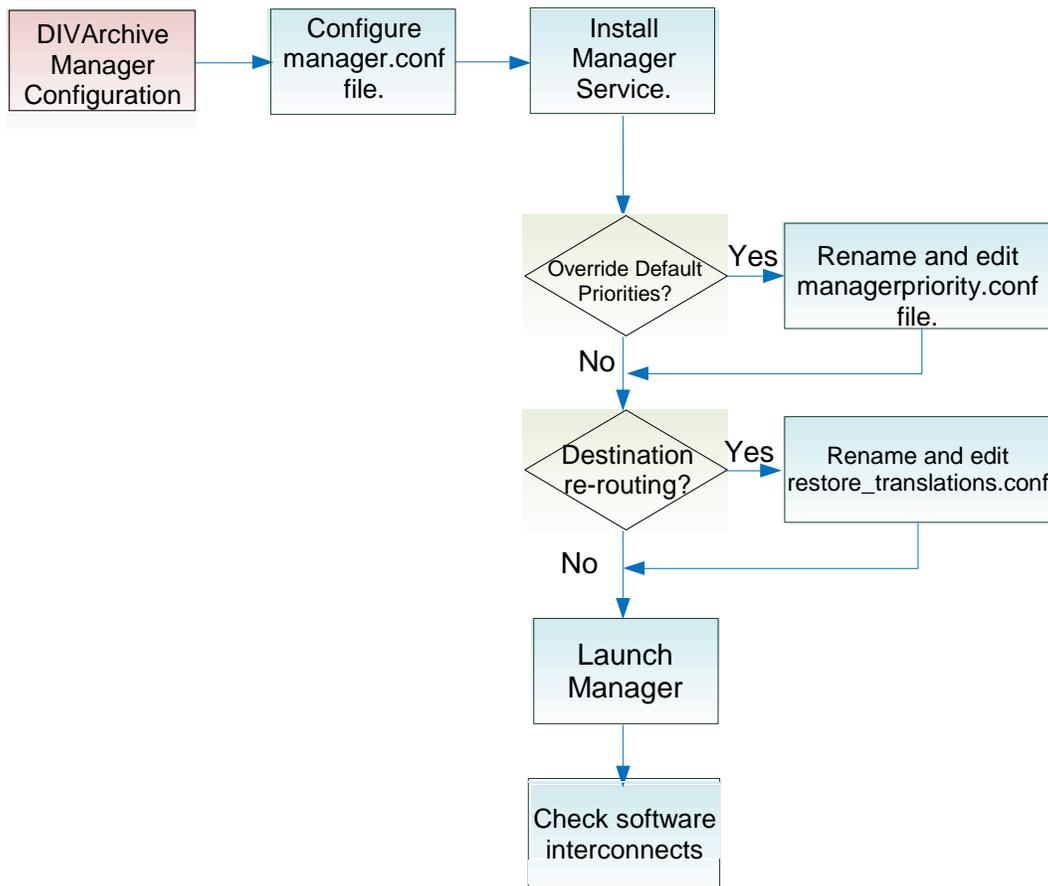
6 DIVArchive Manager Configuration

By default, the Manager module is located in `DIVA_HOME\Programs\Manager\bin` and runs as a Windows Service. The static configuration file for the Manager is `manager.conf`. Most settings in this file can usually be left set to the default. The settings that would normally require updating are highlighted in bold type in 6.2

6.1 Configuration Overview

The following figure depicts the workflow for configuring the DIVArchive Manager.

Figure 130: DIVArchive Manager Configuration Overview



6.2 Local DIVArchive Manager Configuration (`manager.conf`)

The static configuration file for the DIVArchive Manager is `manager.conf`, located in the `DIVA_HOME\Program\conf\manager` folder. In new installations, it is initially named `manager.conf.ini`. The `.ini` extension must be removed for it to be recognized by the DIVArchive Manager.

The configuration file is divided into five distinct groups: **Basic**, **Database**, **Advanced**, **Logging**, and **Service Settings** and are covered in the following sections. The Service Settings section is not intended to be modified in the field and thus not covered in this manual. Values defined in this section should only be altered under instruction of Oracle Technical Support.

Each parameter section in the configuration file contains useful information on defining that parameter. These lines are preceded by a hash symbol (`#`) and are ignored by the Manager. Any parameter definition that is missing the equal sign (`=`) is also ignored.

Note: Spaces in the parameter settings are significant. Do not put extra spaces before or after the parameter names or their values. If you have trouble running the Manager after configuring the `manager.conf` file, double check that superfluous spaces are not present in any of the parameter values have been defined.

Restarting the Manager can be quite disruptive to a live production system and most of the customizations in the configuration file can be made effective immediately by using the **reload** option from the command line.

For users intending to update their existing DIVArchive System with a newer software release, the `manager.conf.ini` from the new installation should always be used with the basic and database settings updated from the old configuration file. The configuration file may have additional settings or updates applied to it (*this applies to all DIVArchive software modules when applying an update*).

6.2.1 New and Updated Parameters Added to `manager.conf`

All of these parameters are reloadable in SERVICE Mode.

Table 31: New `manager.conf` Parameters

Parameter	Description
<code>DIVAMANAGER_MAX_SIMULTANEOUS_REQUESTS</code>	The maximum number of simultaneous requests processed by the DIVArchive Manager. Additional requests are rejected. The maximum tested value for this setting is 2000 – the default value is 500.
<code>DIVAMANAGER_REQUEST_SCHEDULING_QUEUE_SIZE</code>	The maximum number of requests that can be queued for processing by <code>DIVAMANAGER_MAX_CONCURRENT_REQUESTS</code> processors of the Request Scheduler. The default value is 500.

Parameter	Description
DIVAMANAGER_API_TASK_QUEUE_SIZE	The number of tasks that will be accepted to the API command processing queue. If this queue is full, subsequent commands will be rejected. The default value is 2000.
DIVAMANAGER_MAX_CONCURRENT_REQUESTS	The maximum number of concurrent requests executed by the DIVArchive Manager. The default value is 8. The maximum tested value for this setting is 16.
DIVAMANAGER_MIN_DB_CONNECTION_LIMIT	The minimum number of database connections available to the DIVArchive Manager. The default value is 1.
DIVAMANAGER_MAX_DB_CONNECTION_LIMIT	The maximum number of database connections available to the DIVArchive Manager. The default value is 10.
DIVAMANAGER_INITIAL_DB_CONNECTION_LIMIT	The initial number of database connections available to the DIVArchive Manager. The default value is 1.
DIVAMANAGER_INACTIVITY_TIMEOUT	The maximum time a physical connection can remain idle in a connection cache before it is terminated in seconds. The default value is 60.
DIVAMANAGER_SIZE_OF_STATEMENT_CACHE	The size of the database statement cache. The default value is 10.
DIVAMANAGER_DEFAULT_ROW_PREFETCH	The default number of rows to prefetch from the database per query. The default value is 1000.
DIVAMANAGER_FAILOVER_ENABLED	Whether or not to enable Fast Connection Failover. This feature introduces a slight performance penalty. The default value is false .
DIVAMANAGER_NUM_RS_SOLUTIONS_TO_EVALUATE	The number of immediate solutions to evaluate per invocation of the Best Solution Finder during resource selection. The default value is 0; which means it is disabled.

6.2.2 Basic Manager Settings

With the exception of the Service Name, parameters in this section are always required and **must** be defined for the Manager to start successfully. They define basic settings on how other DIVArchive Software Components and DIVArchive API Clients connect to the Manager.

Note: These settings are not reloadable while the Manager is running, and require a Manager restart to take effect.

Table 32: Basic Manager Settings

Basic Manager Settings	Parameter Type	Default
SERVICE_NAME	Name	
This can be used to specify the name of the Windows Service. If not defined, the Service Name defaults to DIVArchive Manager .		
DIVAMANAGER_NAME =	Name	DIVA
The Name this instance of DIVArchive Manager uses to identify itself to other DIVArchive Managers that share its resources but otherwise is arbitrary. It must be unique in a system running multiple Managers with the exception of Main and Backup Managers (<i>configured as a cold standby</i>). In this instance, the Name should be identical.		
DIVAMANAGER_PORT =	TCP Port Number	9000
The TCP Port other DIVArchive applications, or third party applications that use the DIVArchive Client API will utilize to connect to the DIVArchive Manager. If using a Sony Library and running the Manager on the same machine as the PetaSite Controller (<i>PSC</i>) software, be aware that the PSC Server also uses TCP port 9000 and that this cannot be modified. In this situation, another port must be specified for the Manager.		

6.2.3 Database Settings

These parameters define the location and instance of the DIVArchive Database. With the exception of the `DIVAMANAGER_TNSNAME` parameter, all settings in this section must be defined for the DIVArchive Manager to launch successfully.

Table 33: Database Settings

Database Settings	Parameter Type	Default
<code>DIVAMANAGER_TNSNAME</code>	Name	
<p>TNS Name of the DIVArchive Schema within the Oracle database. This setting is ignored if the <code>DIVAMANAGER_DBHOST</code> and <code>DIVAMANAGER_DBPORT</code> settings below are defined.</p> <p>Note: This feature requires Oracle 11g or higher installed on the host running the Manager. If this setting is defined, the location of the Oracle OCI driver (e.g. <code>ocijdbc11.dll</code>) must be added to the <code>wrapper.java.library.path</code> setting (<i>located in Service Settings section of this file</i>); otherwise the Manager will not start as a Windows service.</p> <p>Example: <code>wrapper.java.library.path=.;C:\app\oracle\product\11.1.0\BIN</code></p>		
<code>DIVAMANAGER_DBHOST</code>	IP Address or Hostname	
<p>This specifies the Hostname or IP Address of the machine containing the DIVArchive Database. If using a hostname, this must be present in the <code>hosts</code> file on the machine where the DIVArchive Manager is installed.</p>		
<code>DIVAMANAGER_DBPORT</code>	TCP Port Number	1521
<p>The Oracle Listener port configured during the DIVArchive Database installation.</p>		
<code>DIVAMANAGER_DBSID</code>	Name	11b5
<p>The DIVArchive Database Instance System Identifier (<i>SID</i>) in Oracle where DIVArchive Manager connects. Typically <code>11b5</code> in most DIVArchive installations. Consult your delivery plan if you are not sure.</p>		
Database Settings	Parameter Type	Default
<code>DIVAMANAGER_DBUSER</code>	Name	diva
<p>User Name the DIVArchive Manager uses to connect to the DIVArchive Database. This is typically <code>diva</code> (<i>case sensitive</i>).</p>		

Database Settings	Parameter Type	Default
DIVAMANAGER_DBPASSWORD	Name	lib5
Password for the DIVArchive User in Oracle.		

6.2.4 Advanced Settings

Parameters in this section are typically left at their defaults. They can be used to customize DIVArchive's default behavior for task execution, resource allocation, and the number of connections it will accept from DIVArchive Applications and DIVArchive API Clients. These parameters are normally adjusted or fine-tuned after completing the initial installation of DIVArchive.

Most (*but not all*) of these settings can be altered while the Manager is running by using the **reload** option.

Table 34: Advanced Settings

Advanced Settings	Parameter Type	Default
DIVAMANAGER_TO_LOWER	true/false	false
Sets case sensitivity for DIVArchive. If set to true, then all Object Names, Categories and Groups will be set to lowercase.		
DIVAMANAGER_MAX_CONNECTIONS	Number of connections	200
Specifies the maximum number of simultaneous client connections the Manager will accept. This includes DIVArchive Actors, Control GUI's, API connections and miscellaneous support tools.		
DIVAMANAGER_MAX_SIMULTANEOUS_REQUESTS	Number of requests	500
The maximum number of requests processed by the DIVArchive Manager. When this limit is reached, any further requests will be rejected. The maximum tested value for this setting is 2000.		
DIVAMANAGER_MAX_INACTIVE_REQUESTS	Number of requests	0
Maximum number of inactive requests that cannot find resources examined by the Request Scheduler each time it is activated.		

Advanced Settings	Parameter Type	Default
DIVAMANAGER_TYPICAL_OBJECT_SIZE	Percentage	10
<p>Under normal circumstances, a DIVArchive Actor retrieves the file size of an object prior to an archive transfer and this value is used to determine where the file can be best located on the tape. Some servers do not indicate the file size of an object prior to a Direct Archive so DIVArchive will use this value as an estimate for tape selection. This setting should be defined so that most objects to be archived in the DIVArchive System are below this size. The default value is 10 (<i>for 10%</i>).</p>		
DIVAMANAGER_MAX_SPAN_SEGMENTS	Number	2
<p>If no more writable tapes with enough free space are available to archive a file, DIVArchive will attempt to span the file across 2 or more tapes. This setting defines the maximum number of tapes across which the object will be spanned. The default value is 2 (<i>for 2 segments</i>).</p>		
DIVAMANAGER_CAPACITY_LOW_WATER_MARK	Percentage	90
<p>When the percentage of the total capacity used reaches this amount, periodic warning messages will be issued in the Control GUI to warn the operator. By default, this setting is 90%.</p>		
DIVAMANAGER_ENABLE_SPANNING_LARGE_OBJECTS	true/false	true
<p>Enables spanning of large objects. This parameter overrides <code>SPAN_SEGMENTS</code> if any object in the system is known to be too large.</p>		
DIVAMANAGER_MAX_OBJECTS_FOR_REPACK	Number	500
<p>Repacking a tape with a particularly large number of objects can consume resources for a lengthy period without reclaiming a great deal of unused space in the process. This setting limits the selection of tapes in manual and automatic repacks based on the number of objects to prevent this condition.</p>		
DIVAMANAGER_STOP_IMMEDIATELY_FOR_REPACK	true/false	true
<p>At the end of the interval for Automatic Tape Repack, this setting specifies whether to complete any repack requests still running or to abort them. If this is set to true, repack requests still in progress at the end of the Automatic Repack Window will be aborted.</p>		

Advanced Settings	Parameter Type	Default
DIVAMANAGER_DISMOUNT_AFTER	Time in milliseconds	120000
<p>This specifies the time in milliseconds to automatically dismount a mounted tape no longer needed by any other request. The default is 120,000ms or 2 minutes.</p>		
DIVAMANAGER_UPDATE_PRIORITIES_PERIOD	Time in milliseconds	60000
<p>DIVArchive periodically examines all requests in its request queue and increments their request priority. This prevents a condition where low priority requests may be continually superseded by higher priority requests. This setting specifies the time between updates of the queue by the Manager. The default value is 60000ms or 1 minute. To disable priority updates, set this value to 0.</p>		
DIVAMANAGER_MAX_DELAY_BETWEEN_SCHEDULER	Time in milliseconds	5000
<p>The maximum number of milliseconds between two Request Scheduler Activations when the Manager is always busy. Default value is 5000ms or 5 seconds.</p>		
DIVAMANAGER_SCHEDULER_AFTER_INACTIVITY	Time in milliseconds	500
<p>The number of milliseconds after which a requested Request Scheduler Activation can be launched and if the manager has nothing else to do. This duration should be significantly lower than DIVAMANAGER_MAX_DELAY_BETWEEN_SCHEDULER. This value should not need modification.</p>		
DIVAMANAGER_PING_INTERVAL	Time in milliseconds	600000
<p>This defines the interval (<i>in milliseconds</i>) between checks by the Manager to see if the connections to its clients or services are still active (<i>Actors, SPM's, Control GUI's, etc.</i>). Default value is 600,000ms, or 10 minutes.</p>		
DIVAMANAGER_EXPORT_ROOT_DIR	Directory Path	exported
<p>The Export Tapes feature allows the sharing of DIVArchive Tapes between two or more separate DIVArchive platforms. This setting defines the root folder for the Metadata files for the exported tapes. The folder should exist on the host where the DIVArchive Manager is running and have write permissions enabled.</p>		

Advanced Settings	Parameter Type	Default
DIVAMANAGER_MAX_RESTORE_SERVERS	Number (2 ~ 200)	5
The maximum number of servers allowed in an N-Restore request by a DIVArchive Actor. Maximum value is 200.		
DIVAMANAGER_MAX_EXPORT_TAPES	Number (1 ~ 25)	10
The maximum number of tapes allowed in export request. The maximum value for this setting is 25		
DIVAMANAGER_MAX_EXPORT_ELEMENTS	Number (1 ~ 100000)	100000
This is the maximum number of elements that can be exported using the Export command. The maximum number is 100,000.		
DIVAMANAGER_MAX_FILES_IN_ARCHIVE	Number (1 ~ 1000000)	1000000
The maximum number of files allowed in an archive request. The maximum value is 1,000,000.		
DIVAMANAGER_MAX_FILES_IN_PARTIAL_RESTORE	Number (1 ~ 1000000)	1000000
The maximum number of files allowed in a partial file restore request. The maximum value is 1,000,000.		
USE_IMPROVED_BEST_WORST_FIT_ALGORITHM	true/false	true
When a file was archived to tape in older versions of DIVArchive, the Best/Worst Fit algorithm would select the tape with the largest remaining free size. This could result over time in a low number of blank tapes for tape repacking etc. The new algorithm now selects the tape based on smallest free space and in turn fills all tapes prior to using more free tapes. The new algorithm is now used by default.		
DIVAMANAGER_SITE_SUPPORT_ENABLED	true/false	false
Resources within DIVArchive can be defined by their location. If this parameter is set to true, the Manager will first try to perform the request from the sites specified as MAIN , and if unsuccessful will retry the request with resources from all other sites. The default value is false (<i>sites are ignored and all site resources are considered equally</i>).		

Advanced Settings	Parameter Type	Default
<code>DIVAMANAGER_CACHE_QOS_USE_DISK</code>	true/false	true
<p>In the earlier versions of DIVArchive, a restore operation with a Quality of Service of <code>CACHE</code> or <code>CACHE and DIRECT</code>, would result in the Tape Instance being used as first priority, even if there was a Disk Instance. This setting instructs DIVArchive to use the Disk Instance regardless of the QOS method specified.</p>		
<code>DIVAMANAGER_PRIORITY_TIER</code>	Number (0 – 100)	0
<p>DIVArchive bases the execution of requests in its request queue by its priority number, but there can be instances where a request in the queue with lower priority utilizes a tape that is already mounted. Giving this request priority over others lower in the queue can save a substantial amount of time in tape mount/dismount operations and reduce wear and tear on the tape drives.</p> <p>If this setting is enabled, DIVArchive will examine the request queue for lower priority requests that involve a tape that is already mounted in a drive and add the number specified here to the requests priority. For example, if the request priority is 25, and the Priority Tier value is 50, the total request priority will be 75. By default, this feature is set to 0 (<i>disabled</i>).</p>		
<code>DIVAMANAGER_ETC_FEATURE</code>	true/false	false
<p>This enables the Estimated Time to Complete feature. This function over time gathers statistics on the time for completion of all execution states of each DIVArchive request. Setting this value to true enables this feature.</p>		
<code>DIVAMANAGER_ETC_CONFIDENCE_LEVEL</code>	Number	50
<p>Percentage of Slope Confidence Interval for the simple regression statistical function used in the Estimated Time to Complete feature. The default is 50. This setting is ignored if the <code>DIVAMANAGER_ETC_FEATURE</code> is disabled.</p>		
<code>DIVAMANAGER_OVERWRITE_POLICY</code>	Number (0 ~ 2)	1
<p>This value determines how DIVArchive deals with files that already exist on a Destination Server when executing a Restore, Partial File Restore, or N-Restore request as follows:</p> <ol style="list-style-type: none"> 0 If the file to be restored to the destination already exists no overwrite will occur. 1 The Actor will not verify if the files with the same names exist before attempting to overwrite these files. If files with the same names exist a backup of the existing files are made before overwriting the files. 2 The Actor attempts to delete and then write to files with the same names. 		

Advanced Settings	Parameter Type	Default
DIVAMANAGER_OVERWRITE_OVERRIDE	true/false	false
Overrides the policy sent by the external application via a request with the policy set in DIVAMANAGER_OVERWRITE_POLICY .		
DIVAMANAGER_BULK_EXPORT	true/false	true
During export, this setting identifies whether the Manager will export all objects to a single file or create one file per tape. If this is set to true, the export operation will complete faster.		
ATTEMPT_ACCESS_TO_OFFLINE_DISK	true/false	false
If a disk is offline or not visible to all available Actors, the Manager will automatically abort a transfer request for objects residing on that disk. If this setting is enabled the Manager will try the transfer irrespective of disk status.		
CHANGE_DISK_STATE_ON_ERROR	true/false	true
Defines whether the Manager will automatically vary a disk's status to Offline in the event of a transfer error.		
MANAGER_ACTOR_DISK_RETRY_NUMBER	Number	3
If a disk I/O error occurs during a transfer, this setting will set the maximum number of attempts to retry the transfer with alternate Actors that also have access to that disk.		
DISK_STATUS_POLLING_RATE	Number	60000
This defines the rate in milliseconds in which each disk in the system is polled to obtain its total and remaining free space.		
DISK_BUFFER_SPACE	Number	0.05
This defines the percentage of the overall space of a disk to keep free.		

Advanced Settings	Parameter Type	Default
DISK_CONNECTION_STATE_RESET_DELAY	Minutes	1.0
A disk connection will be reset from the Out of Order state when a successful access is completed and this amount of time has passed since the connection was set to Out of Order.		
COMPONENT_SIZE_CONVERSION_TO_KB_RULE	Number	3
When an element is successfully transferred to tape or disk the Actor reports the size of the element in bytes. This value is then converted to Kb before it is saved to the database. The conversion may be one of three possible values:		
<ol style="list-style-type: none"> 1. KBytes = $(bytes / 1024) + 1$ 2. KBytes = $bytes/1024$; but if ($kbytes < 1$) then $kbytes = 1$ 3. Kbytes = $Math.ceil(bytes/1024)$ 		
DIVAMANAGER_MAX_EXCLUDED_INSTANCES	Number	3
The maximum number of instances excluded from a request that are logged as an event.		
LOGGING_ROOT_LEVEL LOGGING_TRACE_LEVEL LOGGING_SERVICE_LEVEL	DEBUG, INFO, WARN, ERROR, FATAL	INFO
Defines the level of information written to the respective log files.		
DIVAMANAGER_MAX_SPAN_SEGMENTS	Number	2
If no more writeable tapes with enough free space are available to archive a file, DIVArchive will attempt to span the file across 2 or more tapes. This setting defines the maximum number of tapes that the object will span. The default value is 2 (<i>for 2 segments</i>). This setting will completely disable spanning if set to 1 or below. If a span case arises, the Manager will retry the request with a new tape using the old Worst Fit algorithm and the first tape in the attempted span will be marked full. If the second attempt fails, the request will abort.		
DIVAMANAGER_MAX_DB_CONNECTION_ATTEMPTS	Number	10000
The maximum number of allowable attempts to connect to the database.		

Advanced Settings	Parameter Type	Default
DIVAMANAGER_MIN_DB_CONNECTION_PERIOD	Number	1000
The minimum time period (<i>in milliseconds</i>) between connection attempts.		
DIVAMANAGER_MAX_FOLDERS_IN_ARCHIVE	Number	10000
The maximum number of folders allowed in an archive request. Performance degradation may occur for values greater than the default value of 10000. The maximum value is 10000.		
DIVAMANAGER_COMPLEX_OBJECT_THRESHOLD	Number	1000
The maximum number of files allowed before an object is classified as a Complex Object. The maximum value is 10000.		
COMPONENT_SIZE_CONVERSION_TO_KB_RULE	Number	3
<p>This is the Object Size Conversion Rule.</p> <p>Use one of the following rules to convert an object component size from Bytes to Kbytes:</p> <p>1: <code>kbytes = (bytes/1024) + 1</code></p> <p>2: <code>kbytes = bytes/1024, but if(kbytes < 1) then kbytes = 1</code></p> <p>3: <code>kbytes = Math.ceil(bytes/1024)</code></p>		
DIVAMANAGER_RESTORE_QOS	CACHE_ONLY, DIRECT_ONLY, DIRECT_AND_CACHE, CACHE_AND_DIRECT, NEARLINE_ONLY, NEARLINE_AND_DIRECT	NEARLINE_AND_DIRECT
This identifies the default Quality of Service for Restore.		
DIVAMANAGER_TIME_TO_WAIT_FOR_GRACEFUL_SHUTDOWN	Minutes	1440 (1 day)
The time to allow for a graceful shutdown to complete.		

6.2.5 Logging Settings

Table 35: Logging Settings

Logging Settings	Parameter Type	Default
LOGGING_ROOT_LEVEL LOGGING_TRACE_LEVEL LOGGING_SERVICE_LEVEL	DEBUG, INFO, WARN, ERROR, FATAL	INFO
<p>Defines the level of information written to the respective log files:</p> <p>DEBUG - All messages within the Manager are logged. Log files will grow quite rapidly.</p> <p>INFO - Only general information is logged.</p> <p>WARN - Warning and information messages are logged.</p> <p>ERROR – Only error messages are logged.</p> <p>FATAL – No messages are logged unless the Manager unexpectedly stops.</p>		
LOGGING_MAXFILESIZE	Kilobytes/Megabytes	10MB
<p>When the log file reaches this size, a new file is generated and the old one renamed with appropriate time and date stamps. Older log files are subsequently compressed automatically to zip files at 1-hour intervals.</p>		
LOGGING_LIFETIME	Hours	50
<p>This setting defines how long to maintain trace service and zipped log files before deleting them.</p>		

6.3 Request Priority Configuration File

Since each request submitted to the DIVArchive Manager is placed in its transfer queue, request priorities allow DIVArchive to differentiate between important requests, such as restore requests, over less important events, such as tape repacks, etc.

The request priority is a number from 0 to 100 with 0 being the lowest priority and 100 being the highest. The request priority is normally specified when the request is submitted (either via the Control GUI or the DIVArchive Client API), but can also be altered after the request is submitted with the **Change Priority** command.

The default request priority for each request type is preset within DIVArchive. These default priorities can be overridden at the customer's discretion using the following:

1. In the `DIVA_HOME\Program\conf\manager` folder, rename the `managerpriority.conf.ini` file to `managerpriority.conf`.
2. Edit the `managerpriority.conf` file in a text editor to set the necessary values for each request type (see example below).
3. For the new settings to take effect, reload the Manager configuration using the `reload` option or restart the Manager.

Figure 131: Sample of `managerpriority.conf`

```
# DIVArchive Manager priority configuration file
#
# This file can be used to set the default priority of each type
# of request. This default priority is used when the client (from
# the GUI or through the API) submits a request with the DEFAULT
# priority.
#
# The default priority value for each request is listed below.
#
#   DEFAULT_ARCHIVE_PRIORITY           = 50
#   DEFAULT_RESTORE_PRIORITY          = 70
#   DEFAULT_DELETE_PRIORITY           = 80
#   DEFAULT_PARTIAL_RESTORE_PRIORITY  = 70
#   DEFAULT_COPY_TO_NEW_OBJECT_PRIORITY = 30
#   DEFAULT_COPY_TO_GROUP_PRIORITY    = 30
#   DEFAULT_ASSOCIATIVE_COPY_PRIORITY = 30
#   DEFAULT_REPACK_PRIORITY           = 10
#   DEFAULT_ENTER_PRIORITY            = 50
#   DEFAULT_EJECT_PRIORITY            = 50
#   DEFAULT_EXPORT_PRIORITY           = 50
#
# To change the default priorities:
# - Modify any value in that file. Allowed values are from 0 to 100.
#   0 is the lowest priority, 100 the highest.
# - Rename this file managerPriority.conf by removing the .ini extension.
# - Start or restart the manager.
#
# Note: Do not put space characters around the '=' sign or at the end of
#       priority lines.

DEFAULT_ARCHIVE_PRIORITY=50
DEFAULT_RESTORE_PRIORITY=70
DEFAULT_DELETE_PRIORITY=80
DEFAULT_PARTIAL_RESTORE_PRIORITY=70
DEFAULT_COPY_TO_GROUP_PRIORITY=30
DEFAULT_COPY_TO_NEW_OBJECT_PRIORITY=30
DEFAULT_ASSOCIATIVE_COPY_PRIORITY=30
DEFAULT_REPACK_PRIORITY=10
DEFAULT_ENTER_PRIORITY=50
DEFAULT_EJECT_PRIORITY=50
DEFAULT_EXPORT_PRIORITY=50
```

Note: Regardless of the priority of a request, the Manager will (*by default*) periodically increment the priority of every request that is already in its request queue. This prevents a condition where a low request priority may be continually overridden by higher priority requests and thus never executed.

This feature can be disabled setting the `DIVAMANAGER_UPDATE_PRIORITIES_PERIOD` parameter in the Manager Configuration File to 0 and reloading the Manager configuration or restarting the Manager.

6.4 Destination Re-routing (restore_translations.conf)

To simplify production workflows DIVArchive can be configured to automatically override the original destination specified in a restore, partial file restore or N-restore request based on the object's category and original destination. This is known as *Destination Re-routing*. Typically this function is used to allow selective transcoding based on an Object's Category.

Configuration of Destination Re-routing is performed by editing the `restore_translations.conf` file. This is co-located with the Manager Configuration File (`manager.conf`) in the `DIVA_HOME\Program\conf\manager` folder.

Note: By default, the `restore_translations.conf` file is delivered with an `.ini` extension. This extension must be removed for this file to be considered by the Manager.

Figure 132: Destination Re-routing (restore_translations.conf)

The image shows a snippet of the `restore_translations.conf` file with several callout boxes explaining the format and logic of the entries.

Callout 1: All re-routing entries must be in the format `DT_x`. `x` can be any value but must be unique amongst all entries.

Callout 2: Destination in a restore request for this rule to apply.

Callout 3: If the Object Category of the request also matches, then the Destination will be re-routed.

Callout 4: New Destination for the restore request.

Code Snippet:

```
matrix
start with DT_x= Any unique numeric value of x is OK
are supported
category_1;Translated_destination_1
; Category 2 ; translated destination 2
#
# skipped 1
#
DT_0=PROFIL
DT_1=SEACHANG, MPEG; PROXY2
DT_100=Destination 3; Category 3 ; TranslatedDestination 3
```

Configuration example:

- A video server accepts clips of Format: Format1.
- The archive contains clips of Format: Format1 and Format2.
- Objects of Format: Format1 have a Category: Cat1.
- Objects of Format: Format2 have a Category: Cat2.

How to set this up:

1. Define a Source (*Source1*) that points to the video server with no restore transcode options.
2. Define another Source (*Source2*) that points to the video server with options to transcode to Format: Format1.
3. Create a `restore_translations.conf` file containing the following line:

```
DT_0=Source1;Cat2;Source2
```

This means: that when an object of Category **Cat2** is restored to Destination **Source1**, re-route it to Destination **Source2** instead. This way, the automation can always use **Source1** as the Destination.

Objects of Format **Format1**, which are directly compatible with the video server, will be restored to **Source1** without transcoding.

Objects of Format **Format2**, which have a Category **Cat2**, will match the configuration line above and will be rerouted to **Source2**, which has options to transcode them to Format **Format1** when restoring.

6.5 DIVArchive Manager Control Options

Control and management functions of the DIVArchive Manager on Windows platforms are performed by utilizing `manager.bat` from a command prompt. The executable is located in the `DIVA_HOME\Program\Manager\bin` folder.

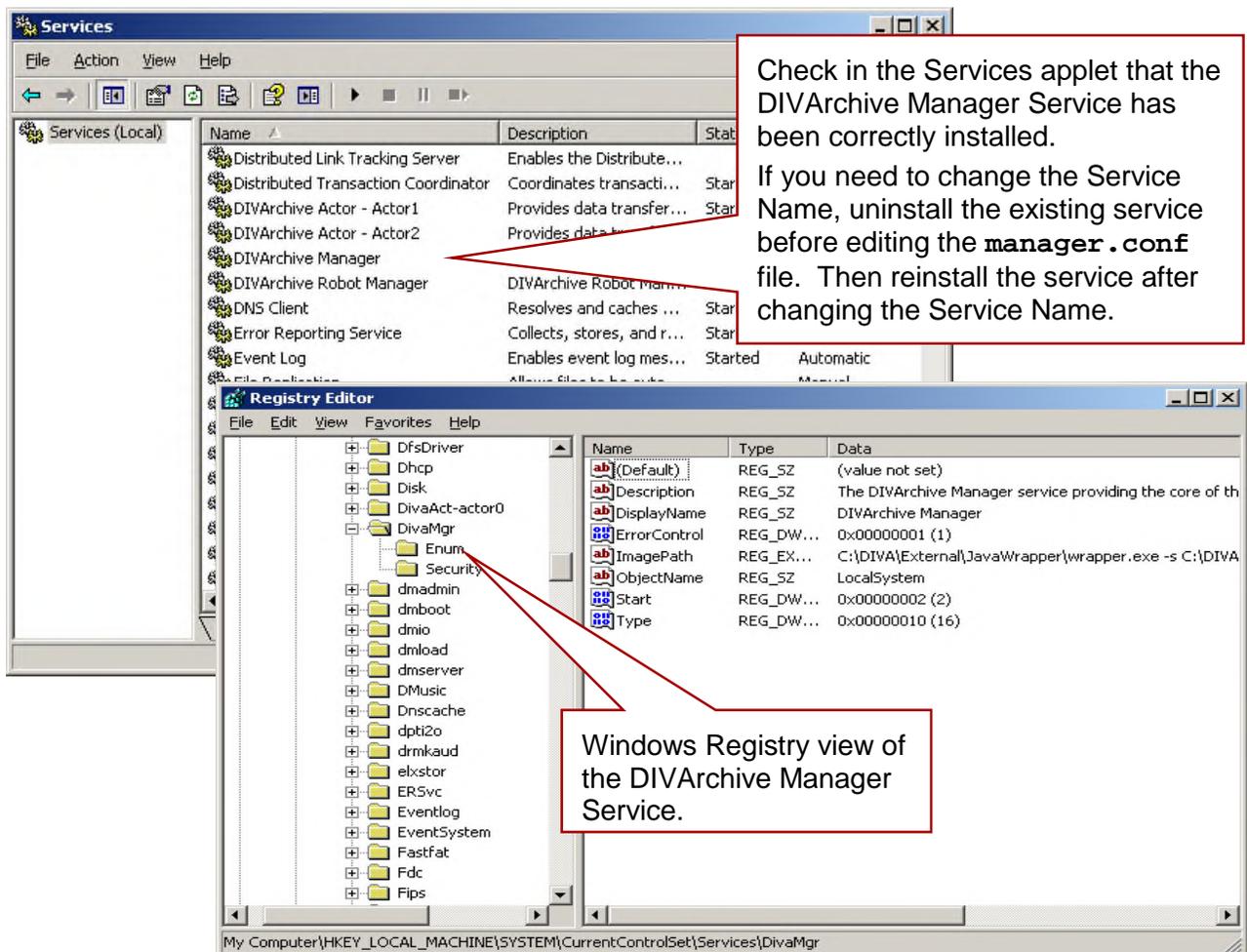
6.5.1 Installing/Removing the DIVArchive Manager Service

On new systems, the DIVArchive Manager must be first installed as a System Service. This is accomplished by using the `-i` and `-u` options:

`manager install (or -i)` Installs the DIVArchive Manager Service set by the `service_name` parameter defined in `manager.conf`. If this parameter is undefined, the service is installed as **DIVArchive Manager**.

`manager uninstall (or -u)` Removes the DIVArchive Manager Service set by the `service_name` parameter defined in `manager.conf`.

Figure 133: Installing/Removing the DIVArchive Manager Service



The default path to the Manager Configuration Files is:

```
DIVA_HOME\Program\conf\manager
```

If an alternate configuration file is to be used other than the default, this can be specified by using the `-conf` or `-f` options:

```
manager install -conf <configuration file>
manager uninstall -conf <configuration file>
```

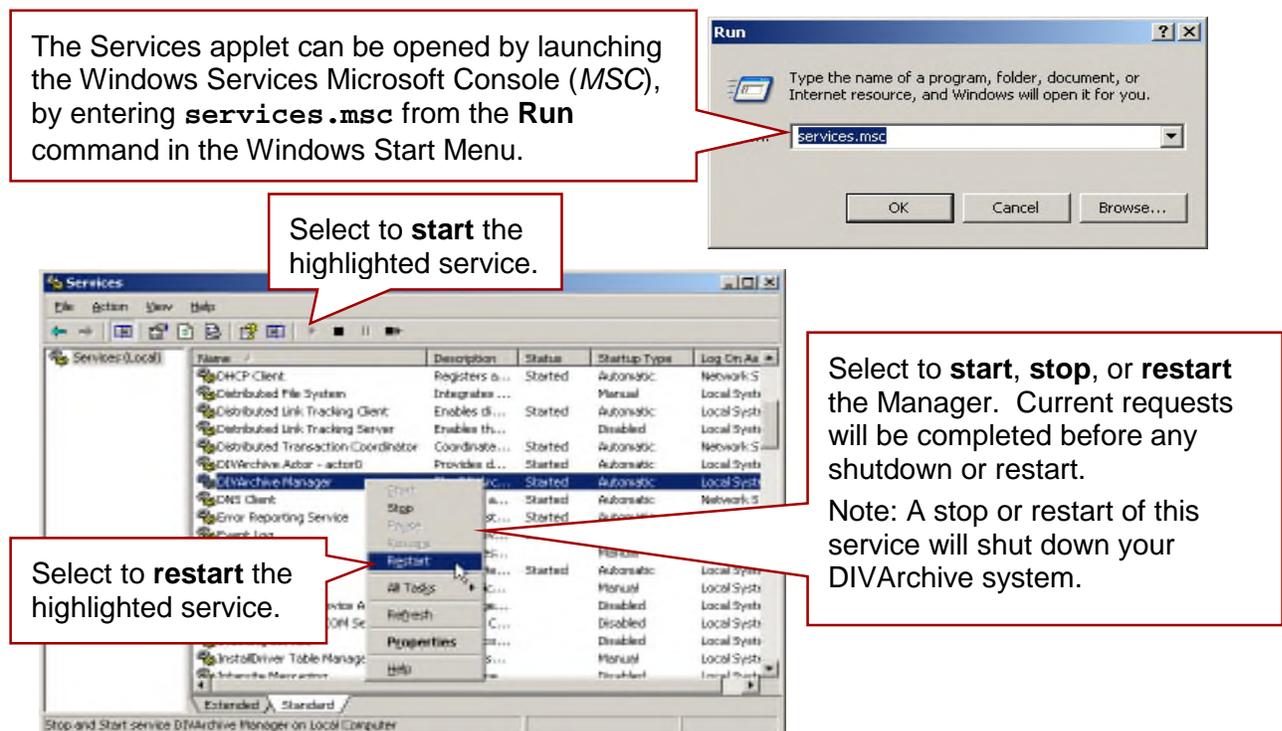
6.5.2 DIVArchive Manager Service Management Functions

The following command options can be used to control the Manager Service after it has been installed:

- manager start** Starts the DIVArchive Manager Service (*if stopped*).
- manager stop** Stops the DIVArchive Manager Service (*if running*).
- manager shutdown** Finishes currently running requests and will stop accepting new requests, then stops the DIVArchive Manager Service (*if running*).
- manager restart** Stops and restarts the DIVArchive Manager Service.

Alternatively, the DIVArchive Manager Service can be managed from the Windows Services applet.

Figure 134: DIVArchive Manager Service Management Functions



Some changes to the Manager Configuration Files will become effective after using the `reload` option:

```
manager reload
```

- The DIVArchive Manager will reload the `manager.conf`, `managerpriority.conf`, and `restore_translations.conf` files from the default path `DIVA_HOME\Program\conf\manager`.

```
manager reload -conf <configuration file>
```

- The DIVArchive Manager will reload the specified configuration file only.

```
manager status
```

- Indicates whether the DIVArchive Manager is running or not.

```
manager dump
```

- Requests that the Manager Service create a system dump.

```
manager version (or -v)
```

- Displays the Manager Service version information and exits.

```
manager help (or -h)
```

- Displays the above command line options and exits.

6.6 Logging

For troubleshooting and diagnostics purposes, the DIVArchive Manager keeps detailed logs of its operations and stores them in the `DIVA_HOME\Program\log\manager` folder. You may be asked for some or all of these logs when contacting Oracle Technical Support.

The level and quantity of information that is captured in each log file is determined by the logging settings in `manager.conf` (see 0). Should these need to be altered, the changes can be made effective immediately using the `manager -reload` option, or in DIVArchive 7.3 changed dynamically from the Control GUI. Refer to the *Oracle DIVArchive Operations Guide* for more details.

Starting with DIVArchive version 7.2, class-level logging is supported via the `manager.classLog.properties` file. Any class set to “X” will log at the current Manager Log Level – where “X” will be one of the following:

- TRACE
- DEBUG
- INFO
- WARN
- ERROR
- FATAL

New statics data is generated every 5 minutes listing various Manager performance-related metrics and is collected in a statistics folder.

Note: Class Level Logging overrides the settings in the `manager.conf` file. The purpose of Class Level Logging is to provide the capability for the Oracle Development Team to gather additional logging information when debugging a customer issue. The defaults should never be changed without consulting the Oracle Development Team.

Sample `manager.classLog.properties` file (*this is not the complete file*):

```
# -----  
# DIVA Manager Class Level Log Properties  
# -----  
  
com.storagetek.ComponentNames=WARN  
com.storagetek.compression.CompressionUtilities=WARN  
com.storagetek.diagnostics.Assert=WARN  
com.storagetek.diagnostics.AssertFailure=WARN  
com.storagetek.diagnostics.EventsQueue=WARN  
com.storagetek.diagnostics.InternalFailureException=WARN  
com.storagetek.diagnostics.QueueingAppender=WARN  
com.storagetek.diagnostics.Trace=WARN
```

Figure 135: DIVArchive Manager Logging Window

Once logs have reached the size defined by **LOGGING_MAXFILESIZE** in **manager.conf**, they are renamed with date and timestamps and compressed (zipped).

This trace log file is currently being written by the Manager.

Name	Date modified	Type	Size
2014-01-14--01.00.05	1/14/2014 1:00 AM	Compressed (zipped) Folder	588 KB
2014-01-14--02.00.05	1/14/2014 2:00 AM	Compressed (zipped) Folder	58 KB
2014-01-14--03.00.00	1/14/2014 3:00 AM	Compressed (zipped) Folder	58 KB
2014-01-14--05.00.00	1/14/2014 5:00 AM	Compressed (zipped) Folder	645 KB
2014-01-14--06.01.02	1/14/2014 6:01 AM	Compressed (zipped) Folder	58 KB
2014-01-14--08.00.00	1/14/2014 8:00 AM	Compressed (zipped) Folder	58 KB
2014-01-14--09.00.05	1/14/2014 9:00 AM	Compressed (zipped) Folder	586 KB
2014-01-14--10.00.05	1/14/2014 10:00 AM	Compressed (zipped) Folder	59 KB
2014-01-14--11.00.00	1/14/2014 11:00 AM	Compressed (zipped) Folder	58 KB
2014-01-14--13.00.00	1/14/2014 1:00 PM	Compressed (zipped) Folder	644 KB
2014-01-14--14.00.05	1/14/2014 2:00 PM	Compressed (zipped) Folder	58 KB
connections	1/14/2014 2:14 PM	CSV File	1 KB
manager.service	1/14/2014 2:13 PM	Notepad++ Document	285 KB
manager.status	1/14/2014 2:14 PM		
manager.trace	1/14/2014 12:53 PM		
process.pid	1/10/2014 4:34 PM		

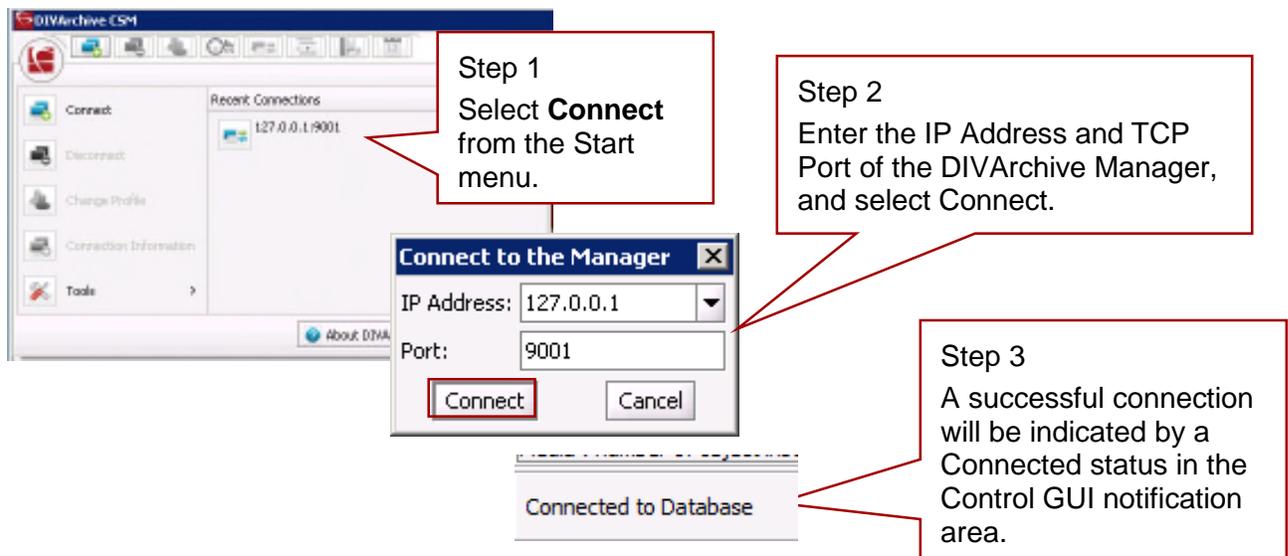
6.7 Check Connectivity

Once the DIVArchive Manager has been successfully configured and launched you should proceed to check that the Manager can be successfully connected to by other DIVArchive clients, such as the Control GUI, and that the Manager itself is able to connect to the configured Actors and, if installed, Robot Managers.

6.7.1 Remote Client to DIVArchive Manager Connection Check

This short test will establish whether the Manager is configured correctly and accepting remote connections from clients. Launch the DIVArchive Control GUI from a remote client (*i.e. not on the same host as the DIVArchive Manager*), and attempt to connect to the Manager.

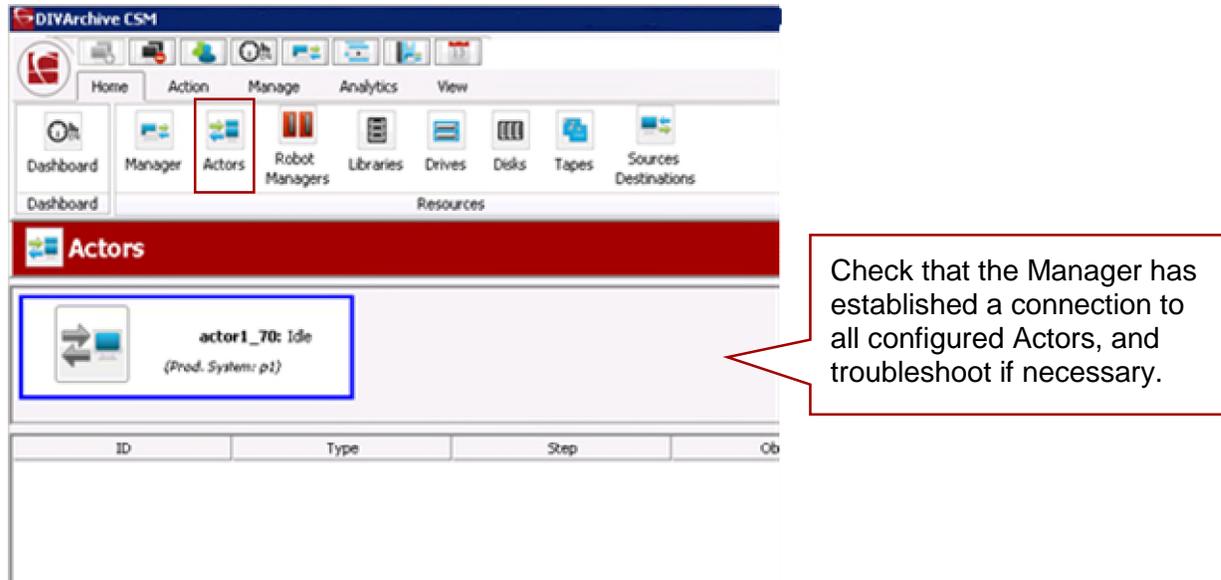
Figure 136: Remote Client to DIVArchive Manager Connection Check



6.7.2 DIVArchive Manager to Actors Connection Check

With the Control GUI still open, select **Actors** in the top icon area to display the Actors view. This test assumes all Actors have been configured correctly and are running.

Figure 137: DIVArchive Manager to Actors Connection Check



6.7.3 DIVArchive Manager to Robot Managers Connection Check

This short test will quickly check if the DIVArchive Manager has connected to each configured DIVArchive Robot Manager.

This test assumes:

1. All DIVArchive Robot Managers have been configured correctly.
2. All libraries are already loaded with tapes.
3. Any library management software (*such as ACSLS*) is running, and the library has been set to **Online**.
4. Each DIVArchive Robot Manager is running.
5. Manual operation has been checked successfully with the DIVArchive Robot Manager Client Tools.

Figure 138: DIVArchive Manager to Robot Managers Connection Check

Select the **Tapes** view in the top icon area.

Note ACS/LSM Number for each tape to test each particular library.

Right-click on a tape for each ACS/LSM to be tested and select **Eject Tape**.

Barcode	ACS	LSM	Media Type	Group	Set	Free space	Tape Format	Fragmentation	Used Capacity	Externalized	Protected	Writable	First Insertion Date	First Utiliza
D00001	0	0	LTO-100G	default	1	893.69 MB	AIF	0%	8%	No		true	18/04/2012 15:14:09	18/04/2012
D00002	0	0	LTO-1			893.38 MB	AIF	0%	8%	No		true	18/04/2012 15:14:09	18/04/2012
D00003	0	0	LTO-1			893.69 MB	AIF	0%	8%	No		true	18/04/2012 15:14:09	18/04/2012
D00004	0	0	LTO-1			893.81				No		true	18/04/2012 15:14:09	18/04/2012
D00005	0	0	LTO-1			976.56				No		true	18/04/2012 15:14:09	18/04/2012
D00006	0	0	LTO-1			976.56				No		true	18/04/2012 15:14:09	18/04/2012
D00007	0	0	LTO-1			976.56				No		true	18/04/2012 15:14:09	18/04/2012
D00008	0	0	LTO-1			976.56				No		true	18/04/2012 15:14:09	18/04/2012
D00009	0	0	LTO-1			976.56				No		true	18/04/2012 15:14:09	18/04/2012
D00010	0	0	LTO-100G			976.56				No		true	18/04/2012 15:14:09	18/04/2012
D00011	0	0	LTO-100G		1	976.56				No		true	18/04/2012 15:14:09	18/04/2012
D00012	0	0	LTO-100G		1	976.56				No		true	18/04/2012 15:14:09	18/04/2012

Check the Manager **Current Requests** view, and double-click on the Eject Tape entry to see if an error was encountered.

ID	Severity	Description	Date
575	Error	Request received : ABORTED	20/04/2012 23:12:59
574	Error	Robot Manager Error	20/04/2012 23:12:59
573	Error	Robot Manager Error : Error while entering tapes (for Robot: robot1_7D(127.D.0.1(8501) : StatusCode[70:INTERNAL_ERROR])	20/04/2012 23:12:59
572	Information	Request step is STEP_WAITING_FOR_OPERATOR (?)	20/04/2012 23:12:59
571	Information	Operator can enter tapes in CAP: CAP0	20/04/2012 23:12:59
570	Information	Request step is STEP_WAITING_FOR_RESOURCES	20/04/2012 23:12:59
569	Information	Request status is PLANNING	20/04/2012 23:12:59
568	Information	Request status is PENDING	20/04/2012 23:12:59
567	Information	Request received	20/04/2012 23:12:59

7 Manager Failover Procedures

In the event of a Manager failure, use the procedures below to switch to a Backup Manager if possible.

Warning: The procedures in this section are critical and sensitive. They should only be performed under the control of an Oracle Support Technician.

Perform these steps on the Main Manager machine:

1. If the DIVArchive Manager Service is still running, try to stop it.
2. Run the "DIVA DB Full backup" scheduled task.
3. Run the "DIVA DB Backup sync" scheduled task.
4. Shutdown the DIVA Manager Server as cleanly as possible.

Perform these steps on the DIVArchive Backup Manager:

5. Change the IP Address to the Main Manager machine's address and reboot the machine.
6. Recover the database:
 - a. Run
`c:\app\oracle\admin\rman\bin\restore_lib5_from_mgr1_to_mgr2.bat`
 - b. Choose **0** for **Automatic Restore** and wait for completion.
 - c. Choose **0** for **Full Backup**.
 - d. Choose **q** to quit.
7. Start the DIVArchive Services (*Manager, RobotManager, Storage Plan Manager, DFM, etc. depending on the configuration*).

The system should now be running from the (*old Backup machine – now Main*) DIVArchive Manager machine allowing repair of the original Main Manager machine.

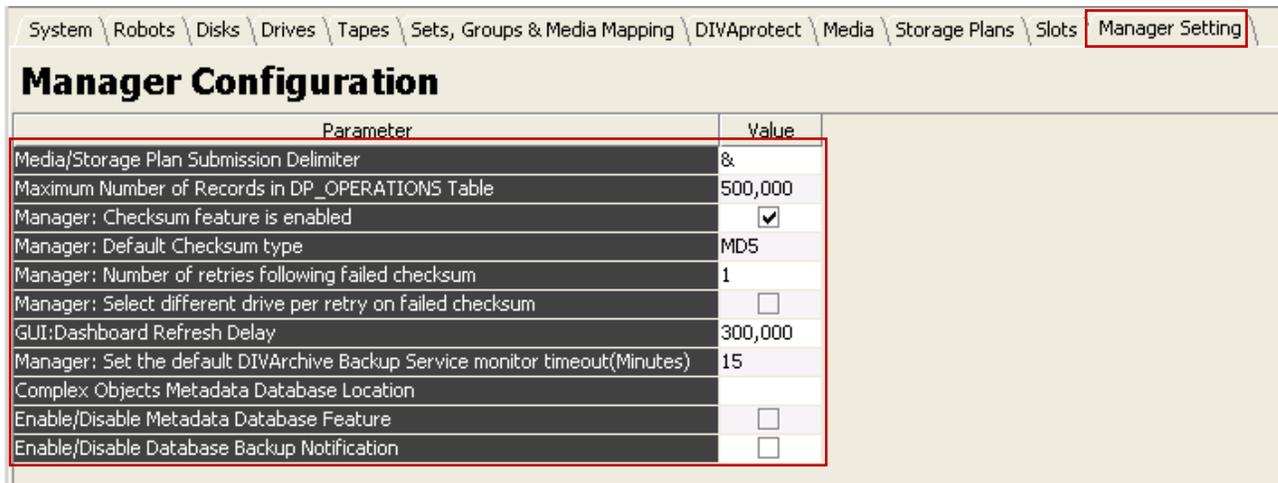
8 DIVArchive Checksum Support Configuration

Configuration of the Checksum Support functions is accomplished through the Configuration Utility using the Engineer Login. The following sections describe how to adjust the settings for each option.

8.1 Global Checksum Parameters

Using the Engineer Login in the Configuration Utility will allow access to the global Checksum Parameters located under the **Manager Setting** tab (outlined in red in the figure below). In the figure below the Global Checksum Parameters are also outlined in red. Each of the Global Parameters affects all Checksum Support settings throughout the system. To view and adjust these setting, the Engineer Login must be used.

Figure 139: Manager Tab of the Configuration Utility and Global Checksum Parameters

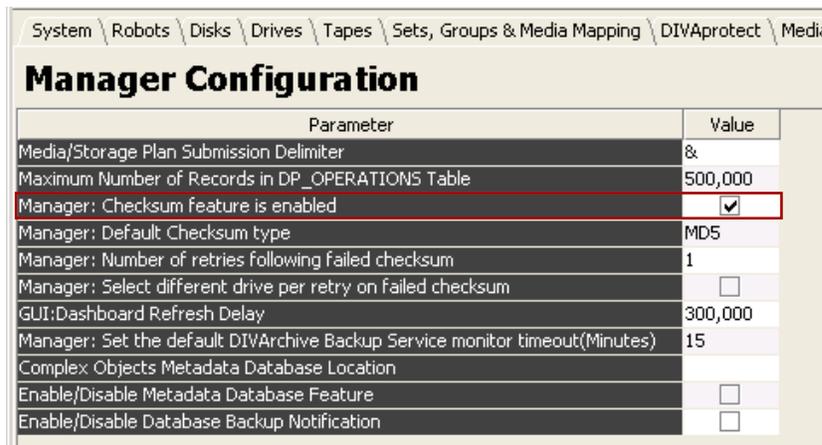


The screenshot shows the 'Manager Setting' tab selected in the Configuration Utility. The breadcrumb trail is: System \ Robots \ Disks \ Drives \ Tapes \ Sets, Groups & Media Mapping \ DIVAprotect \ Media \ Storage Plans \ Slots \ **Manager Setting**. The main heading is 'Manager Configuration'. Below it is a table with two columns: 'Parameter' and 'Value'. The following parameters are listed and highlighted with a red border:

Parameter	Value
Media/Storage Plan Submission Delimiter	&
Maximum Number of Records in DP_OPERATIONS Table	500,000
Manager: Checksum feature is enabled	<input checked="" type="checkbox"/>
Manager: Default Checksum type	MDS
Manager: Number of retries following failed checksum	1
Manager: Select different drive per retry on failed checksum	<input type="checkbox"/>
GUI:Dashboard Refresh Delay	300,000
Manager: Set the default DIVArchive Backup Service monitor timeout(Minutes)	15
Complex Objects Metadata Database Location	
Enable/Disable Metadata Database Feature	<input type="checkbox"/>
Enable/Disable Database Backup Notification	<input type="checkbox"/>

1. **Manager: Checksum feature is enabled:** This setting enables (*checked*) or disables (*unchecked*) the Checksum Support features throughout DIVArchive. The default setting is **on** (*checked*).

Figure 140: Manager: Checksum Feature Is Enabled



The screenshot shows the same 'Manager Configuration' table as in Figure 139, but with the 'Manager: Checksum feature is enabled' row highlighted with a red border. The breadcrumb trail is: System \ Robots \ Disks \ Drives \ Tapes \ Sets, Groups & Media Mapping \ DIVAprotect \ Media.

Parameter	Value
Media/Storage Plan Submission Delimiter	&
Maximum Number of Records in DP_OPERATIONS Table	500,000
Manager: Checksum feature is enabled	<input checked="" type="checkbox"/>
Manager: Default Checksum type	MDS
Manager: Number of retries following failed checksum	1
Manager: Select different drive per retry on failed checksum	<input type="checkbox"/>
GUI:Dashboard Refresh Delay	300,000
Manager: Set the default DIVArchive Backup Service monitor timeout(Minutes)	15
Complex Objects Metadata Database Location	
Enable/Disable Metadata Database Feature	<input type="checkbox"/>
Enable/Disable Database Backup Notification	<input type="checkbox"/>

2. **Manager: Default Checksum Type:** There are several checksum algorithms supported by the system including MD2, MD5, SHA, SHA1, MDC2, and RIPEMD160. MD5 is the default checksum for the DIVArchive System.

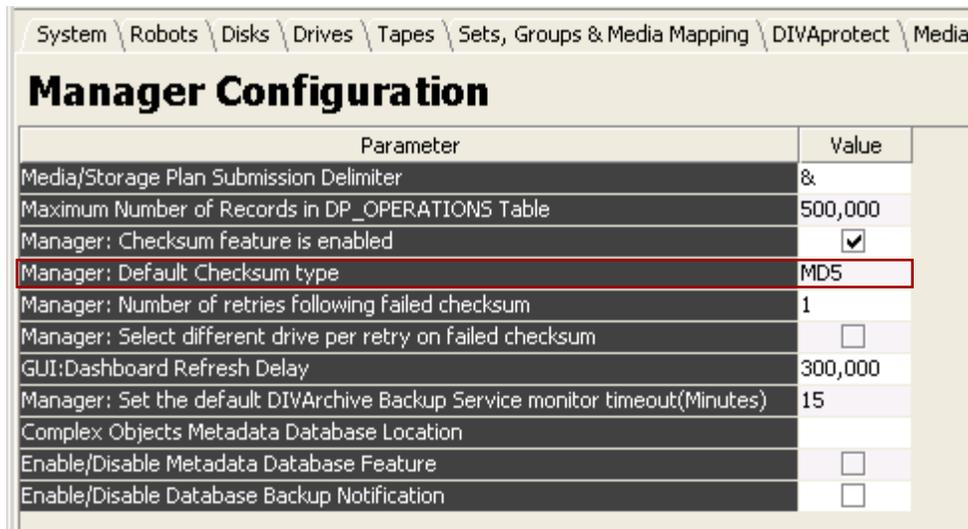
Each Checksum Type is associated with an ID Number as shown in the table below. The ID Number is used to identify the Checksum Type requested in the configuration as follows:

Table 36: Checksum Types and Associated ID Numbers

Checksum Type	ID Number
MD2	1
MD5	2
SHA	3
SHA-1	4
MDC2	5
RIPEMD160	6

To change the default type, use the pull-down menu to select the type of checksum desired.

Figure 141: Manager: Default Checksum Type



3. **Manager: Number of retries following failed checksum:** This parameter sets the number of times the system will retry the operation after a failed checksum. The default setting is one retry. Enter the number of retries allowable for your data and system. **It is recommended that this setting be left at the default value.**

Figure 142: Manager: Number of Retries Following Failed Checksum

System \ Robots \ Disks \ Drives \ Tapes \ Sets, Groups & Media Mapping \ DIVAprotect \ Media	
Manager Configuration	
Parameter	Value
Media/Storage Plan Submission Delimiter	&
Maximum Number of Records in DP_OPERATIONS Table	500,000
Manager: Checksum feature is enabled	<input checked="" type="checkbox"/>
Manager: Default Checksum type	MD5
Manager: Number of retries following failed checksum	1
Manager: Select different drive per retry on failed checksum	<input type="checkbox"/>
GUI:Dashboard Refresh Delay	300,000
Manager: Set the default DIVArchive Backup Service monitor timeout(Minutes)	15
Complex Objects Metadata Database Location	
Enable/Disable Metadata Database Feature	<input type="checkbox"/>
Enable/Disable Database Backup Notification	<input type="checkbox"/>

4. **Manager: Select different drive per retry on failed checksum:** This parameter distinguishes whether the retry (*after a failed checksum*) will be attempted on the same drive (*unchecked*) or if the system should try the operation using a different drive (*checked*). The default setting for this parameter uses the same drive (*unchecked*).

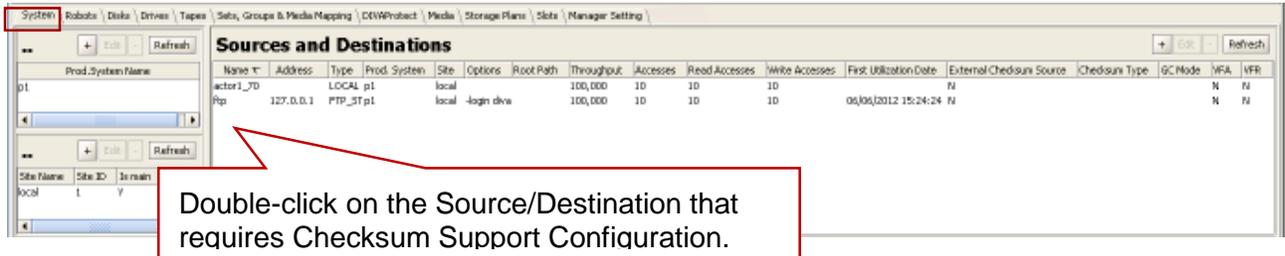
Figure 143: Manager: Select Different Drive per Retry on Failed Checksum

System \ Robots \ Disks \ Drives \ Tapes \ Sets, Groups & Media Mapping \ DIVAprotect \ Media	
Manager Configuration	
Parameter	Value
Media/Storage Plan Submission Delimiter	&
Maximum Number of Records in DP_OPERATIONS Table	500,000
Manager: Checksum feature is enabled	<input checked="" type="checkbox"/>
Manager: Default Checksum type	MD5
Manager: Number of retries following failed checksum	1
Manager: Select different drive per retry on failed checksum	<input type="checkbox"/>
GUI:Dashboard Refresh Delay	300,000
Manager: Set the default DIVArchive Backup Service monitor timeout(Minutes)	15
Complex Objects Metadata Database Location	
Enable/Disable Metadata Database Feature	<input type="checkbox"/>
Enable/Disable Database Backup Notification	<input type="checkbox"/>

8.2 Checksum Configuration for Source/Destination

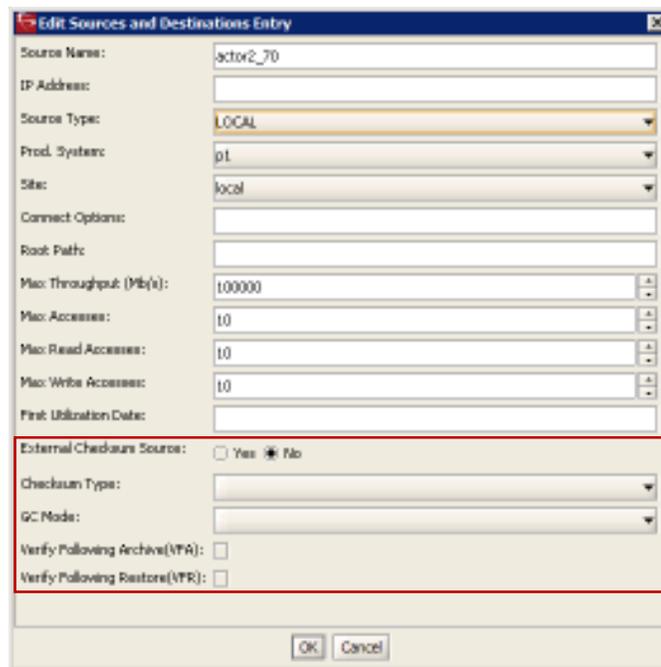
Changing the Checksum Support Configuration for Source/Destinations is accomplished through the Configuration Utility on the **System Tab**. On the right side of the System window display, double-click on the Source/Destination for which Checksum configuration is required.

Figure 144: Source/Destination Area in the Configuration Utility System Tab



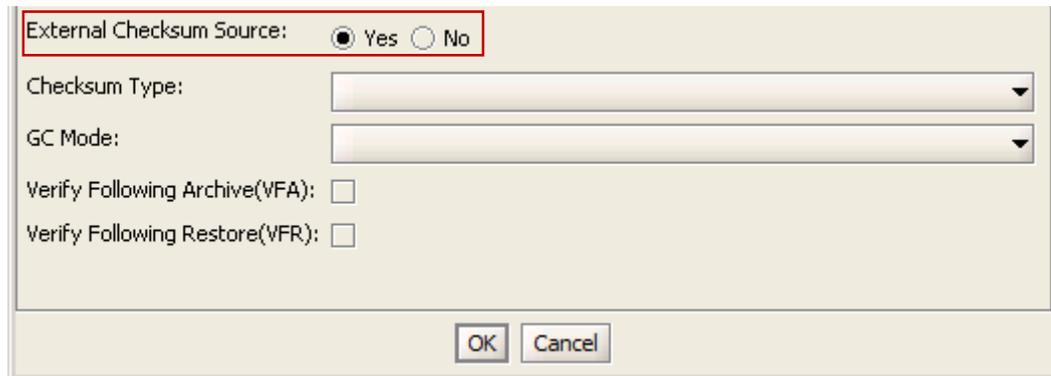
The **Edit Source and Destinations Entry** window (see the figure below) appears with several options pertaining to Checksum Support. These options are mainly associated with the Genuine Checksum Type and outlined in red on the figure.

Figure 145: Edit Sources and Destinations Popup Window



1. The **External Checksum Source** must be used (Yes) in order for the system to read the Checksum from the external source providing the file. This initiates an on-the-fly checksum calculation to compare the checksums and verify the initial transfer.

Figure 146: External Checksum Source Configuration



External Checksum Source: Yes No

Checksum Type:

GC Mode:

Verify Following Archive(VFA):

Verify Following Restore(VFR):

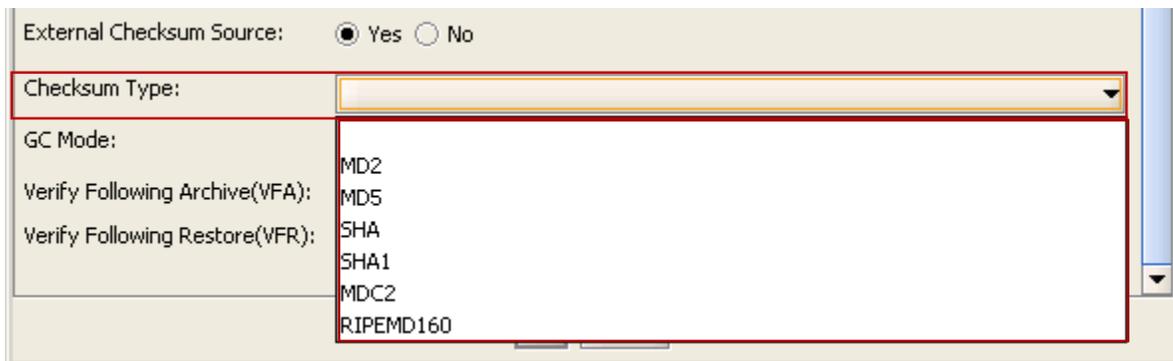
OK Cancel

2. Using the pull-down box select the Checksum Type to use. All supported Checksum Types are listed in the pull-down box. Clicking the mouse while hovering over the desired Checksum Type will select it.

The Genuine Checksum is only used for the first verification. The Checksum Type selected here is only used once and then discarded. Beyond the initial use of this selected Checksum Type (*after this transfer*), the default type is used (*see the figure below*).

The Checksum Type and Mode must match the settings implemented at the Source.

Figure 147: Selecting the Checksum Type Using the Pull-Down Box



External Checksum Source: Yes No

Checksum Type:

GC Mode:

Verify Following Archive(VFA):

Verify Following Restore(VFR):

MD2
MD5
SHA
SHA1
MDC2
RIPEMD160

3. Use the pull-down box to select the Genuine Checksum Mode. This tells the Actor the format of the files that contain the checksum data (see the figure below).

Figure 148: Genuine Checksum Mode Pull-Down Box

External Checksum Source: Yes No

Checksum Type:

GC Mode:

Verify Following Archive(VFA):

Verify Following Restore(VFR):

MDF_XML
TEXT
AXF

OK Cancel

4. When Verify Following Archive (VFA) is turned on (*checked*), performing the initial transfer from the source results in a read-back operation and therefore the data is being read twice for verification. Once the data is read twice, the two checksums are compared. If they are the same then verification is complete, if they are not identical then verification has failed.

Verify Following Archive is not compatible with Genuine Checksum (GC) or Complex Objects. There is no need to use VFA when GC is being used because the checksum is already verified. The Genuine Checksum must be turned off in order to gain access to the VFA checkbox. If GC is turned on, the checkbox will be greyed out and non-clickable.

Figure 149: Verify Following Archive Checkbox

External Checksum Source: Yes No

Checksum Type:

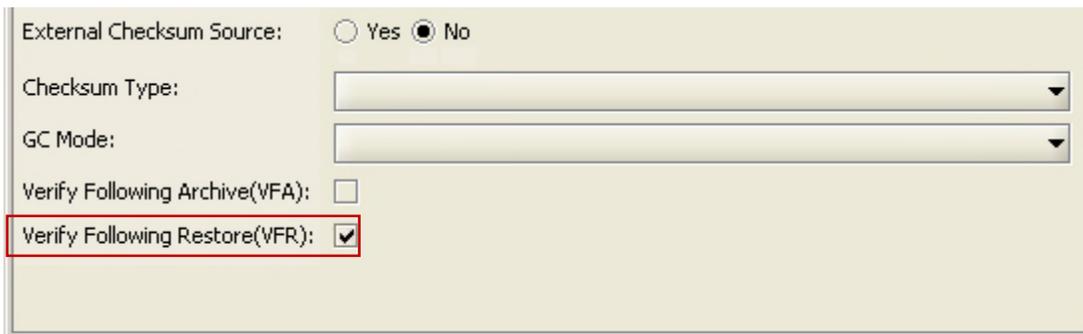
GC Mode:

Verify Following Archive(VFA):

Verify Following Restore(VFR):

5. When Verify Following Restore (*VFR*) is turned on (*checked*), performing the final transfer to the destination results in a read-back operation and the data being read twice for verification. Once the data is read twice, the two checksums are compared. If they are the same then verification is complete, if they are not identical then verification has failed. The setting of GC has no bearing on the VFR setting (*see the figure below*). **VFR is not compatible with Complex Objects.**

Figure 150: Verify Following Restore Checkbox



The image shows a configuration window with the following elements:

- External Checksum Source: Yes No
- Checksum Type:
- GC Mode:
- Verify Following Archive(VFA):
- Verify Following Restore(VFR):

The 'Verify Following Restore(VFR)' checkbox is highlighted with a red border.

Note: Verify Following Restore is not compatible with Complex Objects or the `-axf` option.

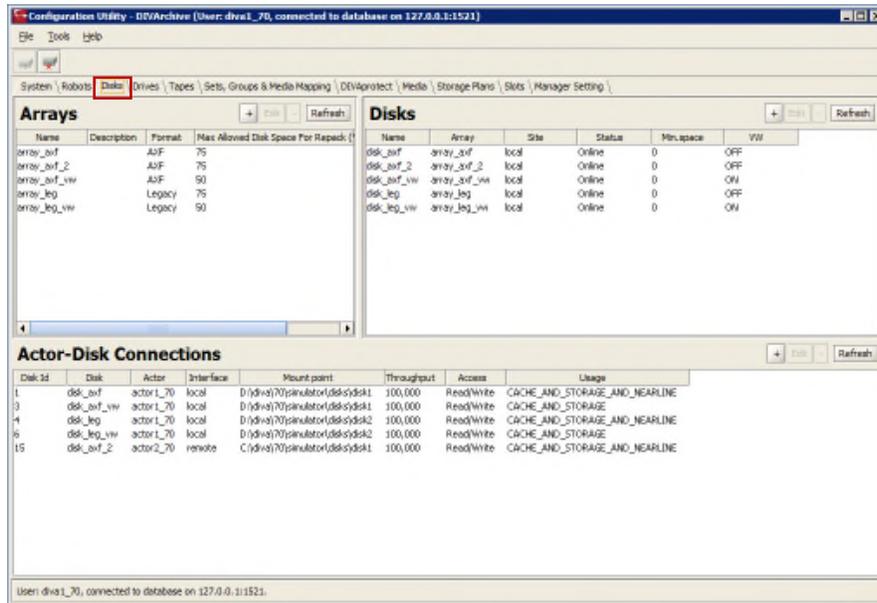
Verify Following Restore was designed to read back the restored content from a video server to confirm that it is not corrupt. Using the `-axf` option does not create a checksum-verifiable restore. It creates an object export that is encompassed in an AXF Wrapper (*container*). These options (*VFR and -axf*) are mutually exclusive and should not be part of the same workflow.

8.3 Checksum Configuration for Arrays and Disks

Changing the Checksum Support configuration for Arrays and Disks is accomplished through the Configuration Utility on the **Disks Tab**. The Verify Write (VW) functionality can be turned on or off either on an array basis or disk-by-disk.

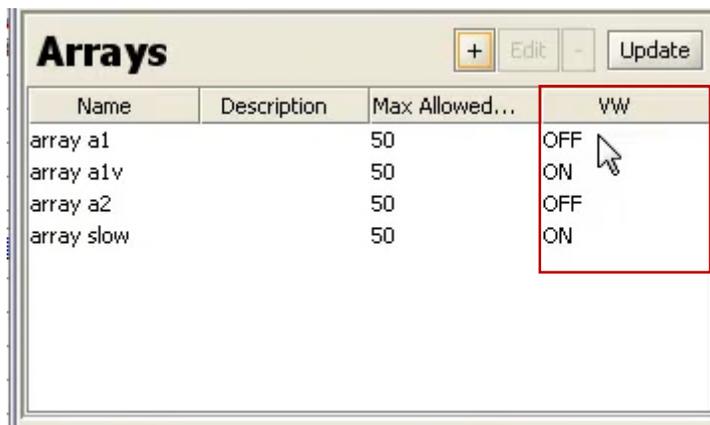
VW applies when you write to the final storage location in DIVArchive. When turned **ON**, the system will perform a read-back of what was just written and compare the checksums for verification.

Figure 151: Arrays and Disks Areas in the Configuration Utility System Tab



The Verify Write (VW) column shown in the figures indicates whether the Verify Write function is on or off for the particular Array and Disk. The default setting is **OFF**.

Figure 152: Verify Write Column in Arrays Area



If there is nothing defined in the VW column on the Disk side of the display, the system will use the setting defined in the Array VW column.

Figure 153: Verify Write Column in Disks Area

Name	Array	Site	Status	Min.space	VW
disk_axf	array_axf	local	Online	0	OFF
disk_axf_2	array_axf_2	local	Online	0	OFF
disk_axf_vw	array_axf_vw	local	Online	0	ON
disk_leg	array_leg	local	Online	0	OFF
disk_leg_vw	array_leg_vw	local	Online	0	ON

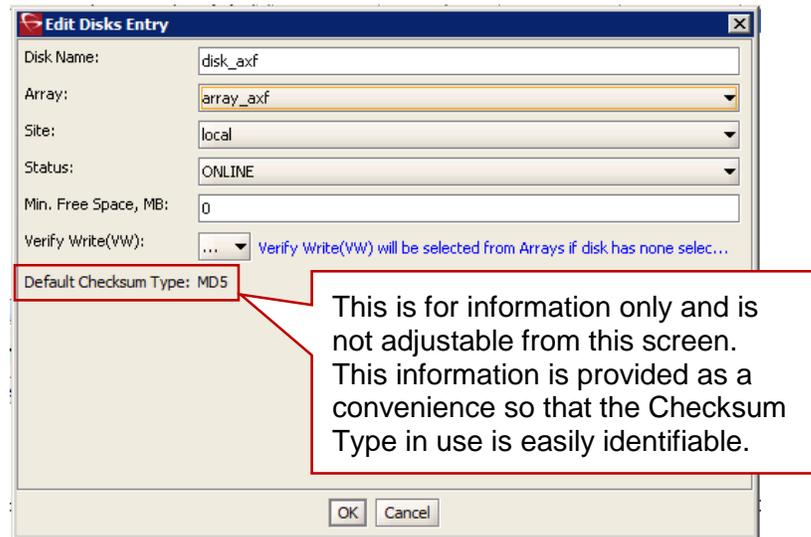
In order to override the setting defined in the Array VW column for a specific Disk, select the Disk that needs configuration and click the **EDIT** button located just above the Disk VW column.

Figure 154: Verify Write Edit Button for Specific Disks



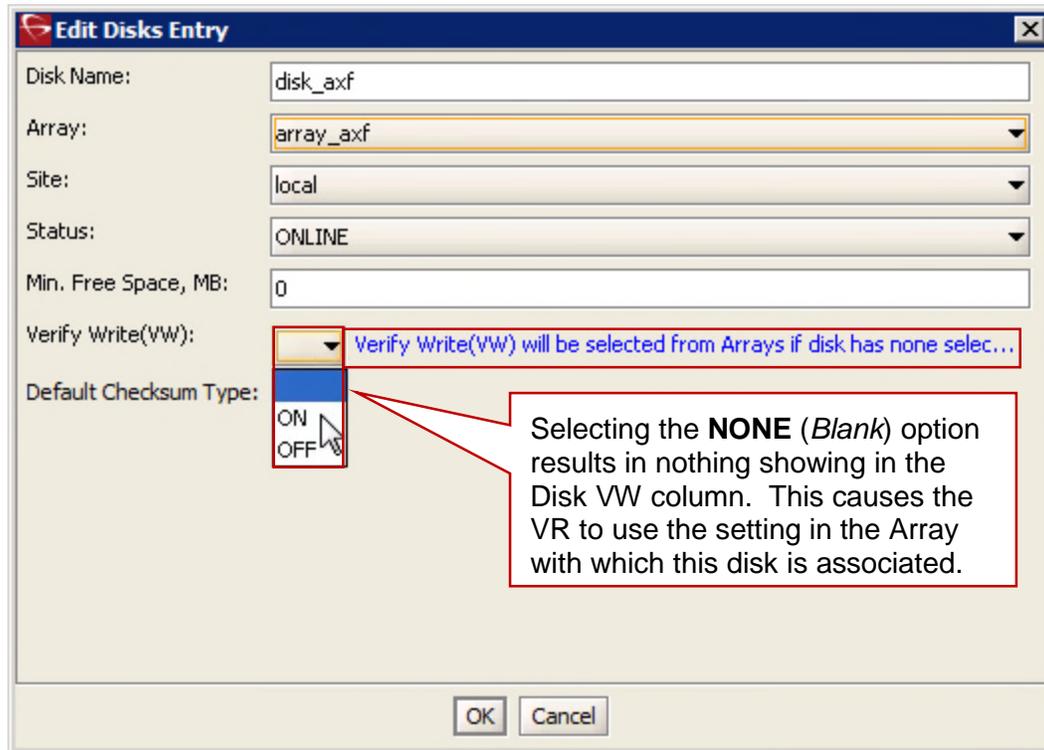
The Edit Row popup window will be displayed as in the figure below.

Figure 155: Edit Row Popup Window



When the Edit Row popup window appears, use the **Verify Write** pull-down to select **ON**, **OFF**, or **NONE** (*Blank selection*). If **NONE** is selected the Verify Write will use the setting identified in the Array for this particular disk.

Figure 156: Verify Write Pull-down Box



The selection made in the Edit Row popup is reflected in the **Disks VW** column. In the figure below, the **ON** selection was made in the popup window's Verify Write pull-down box, and is shown in the Disks VW column.

Figure 157: Disk View in the Configuration Utility Showing Verify Write "On" For this Disk

Name	Array	Site	Status	Min.space	VW
disk_axf	array_axf	local	Online	0	OFF
disk_axf_2	array_axf_2	local	Online	0	OFF
disk_axf_vw	array_axf_vw	local	Online	0	ON
disk_leg	array_leg	local	Online	0	OFF
disk_leg_vw	array_leg_vw	local	Online	0	ON

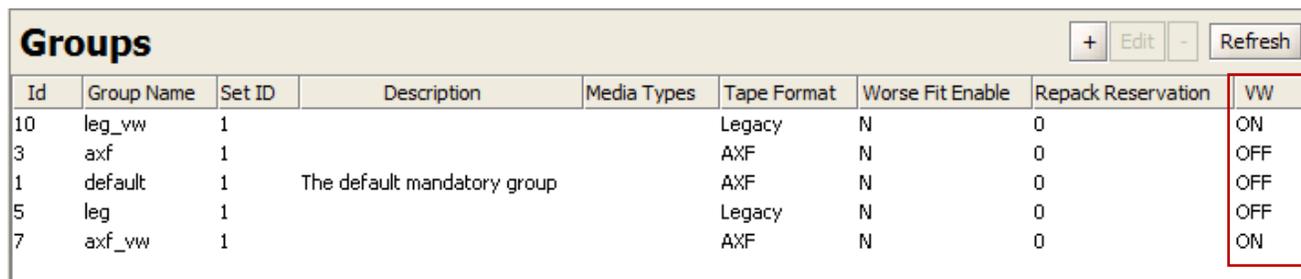
8.4 Checksum Configuration for Groups

Verify Write is also configurable by Groups. Notice the VW column in the Groups display of the Configuration Utility. This is the only place where configuration of Verify Write is available from the Groups.

Similar to the configuration for Disks, select the Group that needs to be configured. Click on the **EDIT** button and select **ON** or **OFF** using the Verify Write pull-down box. The selection made will be reflected in the VW column on the Groups display area as shown in the figure below.

When a file is written to a particular group, the setting for that group will be applied to the file. The default setting is **OFF** for Groups.

Figure 158: Verify Write Column in the Groups Display



The screenshot shows a table titled "Groups" with several columns. The "VW" column is highlighted with a red border. The table contains the following data:

Id	Group Name	Set ID	Description	Media Types	Tape Format	Worse Fit Enable	Repack Reservation	VW
10	leg_vw	1			Legacy	N	0	ON
3	axf	1			AXF	N	0	OFF
1	default	1	The default mandatory group		AXF	N	0	OFF
5	leg	1			Legacy	N	0	OFF
7	axf_vw	1			AXF	N	0	ON

8.5 Checksum Configuration for Actors

Verify Tape is configurable by Actors. Similar to the configuration for Disks and Groups, select the Actor that needs to be configured. Click on the **EDIT** button and select **Yes** or **No** using the Verify Tape pull-down box as shown in the figure below.

This setting defines if the Actor is automatically selected for the Verify Tape workflow. By default, all Actors are included and the operator can exclude if necessary.

Figure 159: Actors Edit Row Popup Window Showing the Verify Tape Pull-down Box

The screenshot shows a window titled "Edit Actors Entry" with a close button in the top right corner. The window contains several configuration fields:

Actor Name:	actor1_70
IP Address:	127.0.0.1
Port:	9901
Prod. System:	p1
Site:	local
Max Drive Operations:	10
Max Server Operations:	10
Max Disk Operations:	10
Verify Tape:	Y
Direct Restore:	Y
Cache Restore:	Y
Copy To Group:	Y
Associative Copy:	Y
Repack:	Y
Delete:	Y
Direct Archive:	Y
Cache Archive:	Y
First Utilization Date:	11/04/2012 20:13:51

At the bottom of the window are two buttons: "OK" and "Cancel". The "Verify Tape" field is highlighted with a red border.

8.6 Oracle SAMMA Solo Integration Configuration

Genuine Checksums are passed from SAMMA to DIVArchive using a XML Metadata File. When the Actor is instructed to read a file using an S/D configured for GC, the Actor generates checksums according to the configuration. These values, as well as the values obtained from the metadata file will be transferred to the Manger for comparison.

For example: for the file `sample.abc`, the metadata filename and structure are as follows:

An XML formatted file with the following parameters as a minimum:

```
<DIVAObjectDefinition>
  <objectName>sample</objectName>
  <fileList>
    <file checksumType="SHA1"
      checksumValue="9F097AAEEF48C4A170D95AAF6161790662626802">
      sample.abc
    </file>
  </fileList>
</DIVAObjectDefinition>
```

The Metadata File Name will be `sample.abc.xml`

There are 3 aspects to the configuration as follows:

1. SAMMA Solo Configuration

Use the following process to configure SAMMA Solo to drop the migrated clip and XML file to specific folder:

1. Start SAMMA Solo.
2. Click **File**, then **New**.
3. Open a `sxt` template (*sxt files are in folder: c:\Program Files\Front Porch Digital\SAMMA Solo*)
4. Log in as the administrator: Click **Configuration**, then **Administrator**, then **Log In**.
5. Click **Configuration**, then **Encoders**.
6. Go through each encoder tab and change the setting for **Move file to here on successful migration**. This is where the migrated clip and XML Metadata File will be dropped by SAMMA Solo (*For example d:\dfm_folder*).
7. Save the `sxt` template for future use by clicking **File**, then **Save**.

8. Place the file `divaXML.vbs` in the folder `c:\Program Files\Front Porch Digital\Common`. This file is provided by SAMMA Solo.
9. Click **Configuration**, then **Options**.
10. Click on the **Commands** tab and set the following command for **On Success**:
 - a. `C:\WINDOWS\system32\cscript.exe`
 - b. `C:\Program Files\Front Porch Digital\Common\divaXML.vbs`
 - c. `f:\Success\$(Settings/Details/Filename).xml`
11. Configure FTP on the DIVAsolo system and add the `d:\dfm_folder` folder so that it is accessible via FTP.

2. DFM Configuration

Operator should install and pre-configure DFM according to the DFM Installation Guide. Use the following process to configure DFM to monitor the specific directory on the SAMMA Solo machine:

In the DFM Configuration File, have a folder type **single** configured to the SAMMA Solo system.

```
<folderConfig>
  <!-- Folder URL. -->
  <url>ftp://diva:diva@172.16.3.47/</url>
  <type>single</type>
  <priority>30</priority>
  <mdfConfigPriority>Primary</mdfConfigPriority>
  <categoryName>dfm_solo</categoryName>
  <incompleteThreshold>86400</incompleteThreshold>
  <sourceDestinationDIVAName>dfm_solo</sourceDestinationDIVAName>
  <archiveFilePathTemplate platform="DETECT"
  options="">URL_TO_FILE</archiveFilePathTemplate>
  <archiveFileNameTemplate platform="DETECT"
  options="">filename.ext</archiveFileNameTemplate>
  <divaMediaName>array_01</divaMediaName>
```

```
<fileFilter type="exclude">
  <mask>*.XML</mask>
</fileFilter>
<deleteBeforeArchive>TRUE</deleteBeforeArchive>
</folderConfig>
```

Table 37: Significance of Highlighted Fields in the Above Example

Field	Significance
<url>	This is the FTP connection details for the SAMMA Solo System.
<type>	Type of folder DFM is monitoring.
<SourceDestinationDIVAName>	Name of the source used when DFM submits the archive request to the Manager.
<divaMediaName>	Media to which DFM will submit the archive request.
<fileFilter>	Used to prevent DFM from archiving the XML Metadata File.

3. DIVArchive Configuration

- Add Source/Destination mentioned in the DFM Configuration File.
- Add Storage Media Name mentioned in DFM Configuration File.
- The Source must be pointing to the SAMMA Solo System FTP.
- The Source must be configured with the following options:

Table 38: Example Data in Reference to DFM Configuration Sample Above

Checksum Type	SHA1
GC_MODE	MDF_XML
divaMediaName	array_01

8.7 AXF and TEXT Genuine Checksum Modes

There are two new Genuine Checksum modes starting with DIVArchive version 7.2:

- TEXT
 - Allows DIVArchive to archive all files and subfolders in a specified folder while comparing their checksum values against known values stored in an External Checksum File.
- AXF
 - Allows DIVArchive to archive all files and subfolders in a specified AXF File while comparing their checksum values against known values stored in the AXF file.

8.7.1 TEXT Genuine Checksum Mode

Requirements:

- A checksum file must be present in the folder specified by the Root File Path.
- Checksum files must end with a .md5 file extension.
- The Checksum Filename (*excluding the md5 extension*) is associated with the folder name which contains all the files that will be archived. This folder must exist.
 - Example: if the Checksum File is **D:\Data\Video\NewTitle.md5** then all files located under the folder **D:\Data\Video\NewTitle** will be archived.
- The Checksum File must be present in the folder parent to the folder specified by the Root File Path.
- For a file to be archived with the Genuine Checksum value, the file must be referenced with a corresponding checksum within the Checksum File.

Configuration in the DIVArchive Configuration Utility:

1. Create a new **Source/Destination** entry with **Source Type** set to either **DISK** or **FTP_STANDARD**.
2. Specify an appropriate **Root Path** – this path, along with the Input Files, specified during the Archive Request will be used in determining the location of the checksum file (see *Selecting the Root File Path below for further details*).

Examples:

- If the Source Type is **DISK**, the Root Path can be set to **D:\Data**.
- If the Source Type is **FTP_STANDARD**, the **Root Path** can be set to **/Data**.

In the DIVArchive Configuration Utility:

1. Set the **External Checksum Source** to **YES**.
2. Set the **Checksum Type** to **MD5**.
3. Set the **GC Mode** to **TEXT**.
4. Click **OK** and go to **Tools>Notify Manager** to notify the Manager of the configuration.

Selecting the Root File Path

The **Root File Path** must point to the folder containing the checksum file. Thus, the correct file/folder paths in the **Source/Destination** and Archive Request form must be set so that the checksum file can be located. For example, if the checksum file is located in `D:\Data\Video\NewTitle.md5` (or `/Data/Video/NewTitle.md5` for *FTP* type), the appropriate file/folder paths can be set as follows:

Table 39: Root File Paths for Disks

Source/Destination (Root Path)	Archive Request (File Path Root)	Archive Request (Files)
D:\	Data\Video\NewTitle	*
D:\Data	Video\NewTitle	*
D:\	Data\	Video\NewTitle*
D:\		Data\Video\NewTitle*

Table 40: Root File Paths for FTP

Source/Destination (Root Path)	Archive Request (File Path Root)	Archive Request (Files)
/	Data/Video/NewTitle	*
/Data	Video/NewTitle	*
/	Data/	Video/NewTitle/*
/		Data/Video/NewTitle/*

Limitations

- DIVArchive cannot open or create files on a Windows filesystem if their absolute path exceeds 260 characters. The Root Path must be no more than a total of 260 characters.
- Only ASCII, non-UTF-8 encoded checksum files are supported.
- The format of the Checksum File is that each line begins with an MD5 Checksum, followed by 2 spaces, and then the File Path to the referenced file.

8.7.2 AXF Genuine Checksum Mode

The addition of the new **AXF Genuine Checksum** mode allows DIVArchive to archive all files and subfolders in a specified AXF File while comparing their checksum values against known values stored in the AXF file. This kind of workflow is typically combined with a Restore Request with **-axf** in the Request Options.

Requirements

- The AXF containing the files to be archived must contain checksum information for each file.
- The AXF must contain checksums of the expected type (*specified in the configuration*).

DIVArchive Configuration Utility Settings

1. Create a new **Source/Destination** entry with **Source Type** set to either `DISK`, `FTP_STANDARD`, or `EXPEDAT` as appropriate.

If required specify an appropriate **Root Path** – this path along with the Input Files specified during the Archive Request, will be used in determining the location of the checksum file.

Examples:

If the **Source Type** is `DISK`, the **Root Path** can be set to `D:\root`.

If the Source Type is `FTP_STANDARD`, the Root Path can be set to `/root`.

2. Set the **External Checksum Source** to **YES**.
3. Set the **Checksum Type** to the expected checksum type (*MD5 for example*).
4. Set the **GC Mode** to **AXF**.
5. Click the **OK Button** and notify the **Manager** of the configuration by selecting **Tools**, then **Notify Manager** from the menu.

Limitations

- This workflow only works with AXF requests generated by DIVArchive.
- Verify Following Restore (*VFR*) is not compatible with the **-axf** option.
 - VFR was designed to read back the restored content from a video server to verify it has not been corrupted. Using the **-axf** option does not create a “real” restore, rather an object export in an AXF Wrapper. These options are mutually exclusive and should not be part of the same workflow.

9 Transcoder Installation and Configuration

The following instructions are directed towards servers running the Windows 2008 Server R2 SP1 Operating System.

9.1 Upgrading from Telestream Vantage Version 5.0 or Earlier

Upgrading from older versions of Vantage (5.0 or earlier) requires uninstalling and reinstalling the Vantage software. Refer to the Vantage 6.3 Installation Guide for details on the uninstall procedure.

9.2 Installing Telestream Vantage

It is recommended that no anti-virus software is installed on the Vantage Servers. Use the following procedure to install Vantage 6.3:

1. Download Vantage release 6.3 from Telestream.
2. If the installer is uncertain of how to install the software, refer to the Quick Start Instructions in the downloaded file.
 - a. Install .NET 3.5 SP1 (*if not already installed*) on the host machine that will be running the Vantage Database Server.
 - i. Server Manager, under “Features”.
3. Install QuickTime 7.6.9 (*if not already installed*).
4. Install the Desktop Experience option.
 - a. Server Manager, under “Features”
5. Install the `vantageDatabaseSetup_SQL2008_4.2.286.100451.exe` (*accept the defaults*).
6. Install the `vantage_6.3_setup.exe` (*accept the defaults*).
 - a. Select the Install Product(s) option.
 - b. Ensure the following options are selected:
 - i. Transcode/Transcode Pro
 - ii. Web Applications
 - iii. Workflow Portal Application
 - iv. Vantage Domain Database
 - c. Enable any other options that will be required for your installation.

9.2.1 Telestream License Installation

1. Once the software is installed, launch the Vantage Workflow Designer.
 - a. If prompted to select a Domain, select the local computer.
 - b. If prompted for a Category click **Cancel** (*for now*).
2. Select **File** and then choose the **Add/Update License**.

Vantage is now installed and can be configured to work with DIVArchive.

It is recommended to import sample workflows within the Vantage Workflow Designer. A demonstration video is available at:

<http://www.telestream.net/vantage/demos.htm>.

9.3 Configuring DIVArchive and Transcoders

The instructions below identify the configuration of DIVArchive and transcoders to allow them to operate together.

Starting with DIVArchive 7.3, it is no longer required to have Actor installed on the same machine as the transcode service.

9.3.1 General Instructions

Use the following procedure to configure Vantage transcoders when the Actor is on the same machine as the transcode service:

1. Create a cache folder on the Actor machine.
 - a. For a Vantage transcoder you could use **M:\VantageCache**.
2. Add the transcoder in the DIVArchive Configuration Utility.
 - a. **Transcoder Type:** vantage
 - b. **Working Directory:** M:\VantageCache
 - c. The remaining options should be left at the default settings.
3. For Vantage, verify that the `vantageTranscoderInterface.dll.config` file is in the `diva_home\program\actor\bin` folder.
4. Ensure that the DIVArchive Transcoder configuration's Simul Transcodes value is less than or equal to the corresponding Vantage Session Limit value.

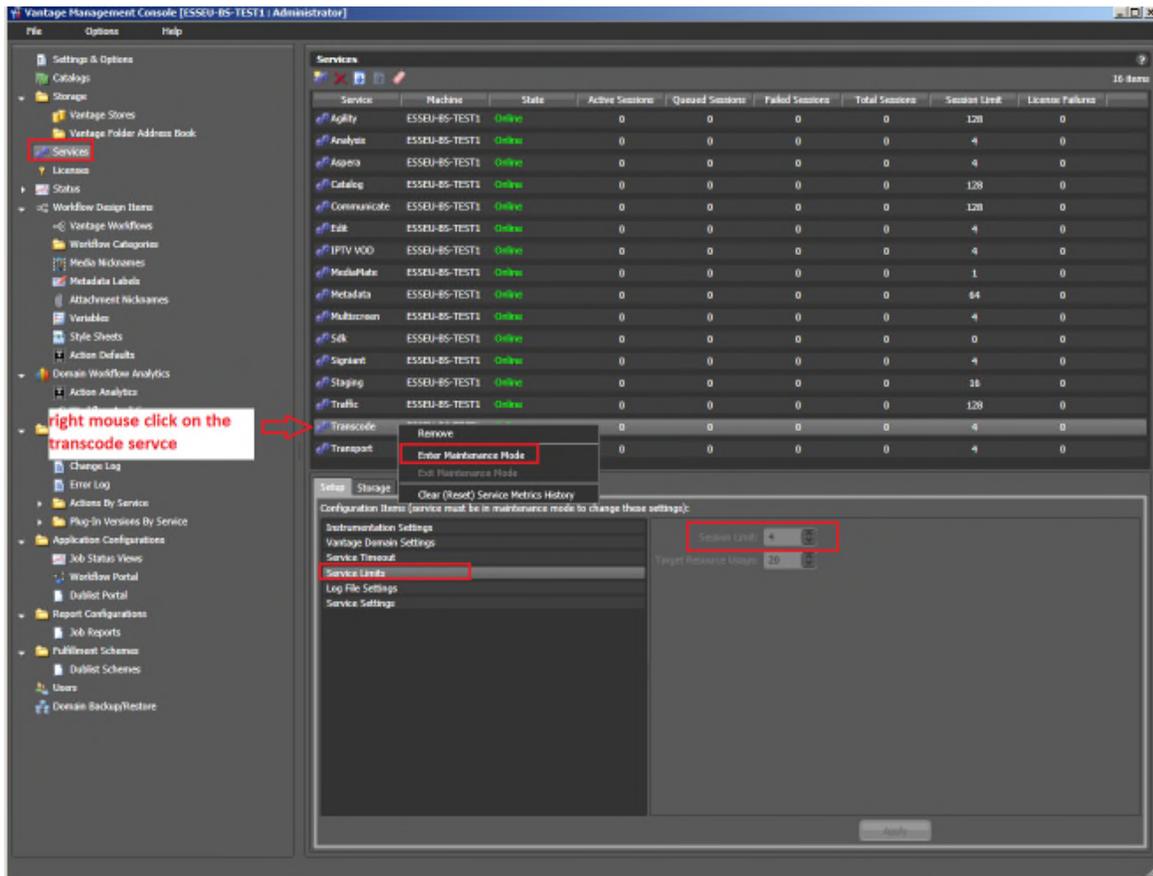
Figure 160: Configuration Utility Transcoder Parameters

Transcoders										
Name	Version	Type	Actor	Transcoder Port	Progress Port	WorkDir	ExePath	Simul Transcodes	Queue Size	Performance
transcoder-i6.3		vantage	actor2	8,676	N/A	D:\Media	Transcoding\RunTCode.exe	4	4	4

9.3.2 Vantage Specific Configuration

The following sections describe only the Vantage transcoder configuration.

Figure 161: Vantage Transcode Limits



Use the following procedure to configure Vantage when the Actor is on a different machine than Vantage Transcode service:

NOTE: The cache folder must be located on the Vantage SDK machine.

5. Create a cache folder on the Vantage SDK machine
 - a. In the example `C:\vantageCache` is used.
 - b. In Windows, share this folder on the network and set the required access credentials for the folder.
6. Add the transcoder to the DIVArchive Configuration Utility.
 - a. **Transcoder Type:** `vantage`
 - b. Set the **Working Directory** as follows:
 - i. Use a CIFS UNC path pointing to the IP address of the Vantage SDK service machine.
 - ii. Set the path to the shared `VantageStorage` folder.
 - c. The remaining options should be left at the default settings.

Figure 162: Edit Transcoders Entry Screen

The screenshot shows a dialog box titled "Add new row in Transcoders". It contains the following fields and values:

Transcoder Name:	vantage_001
Transcoder Version:	null
Transcoder Type:	vantage
Transcoder Port:	8676
Progress Port:	0
Working Directory:	cifs://user:pass@\\172.16.3.27\VatnageStorage
Executable Path:	Transcoding\RunTCode.exe
Simul Transcodes:	21
Queue Size:	51
Performance:	1

At the bottom of the dialog are "OK" and "Cancel" buttons.

7. Verify that the `vantageTranscoderInterface.dll.config` file is in the `diva_home\program\actor\bin` folder.

9.3.2.1 Create the Output Path in Vantage

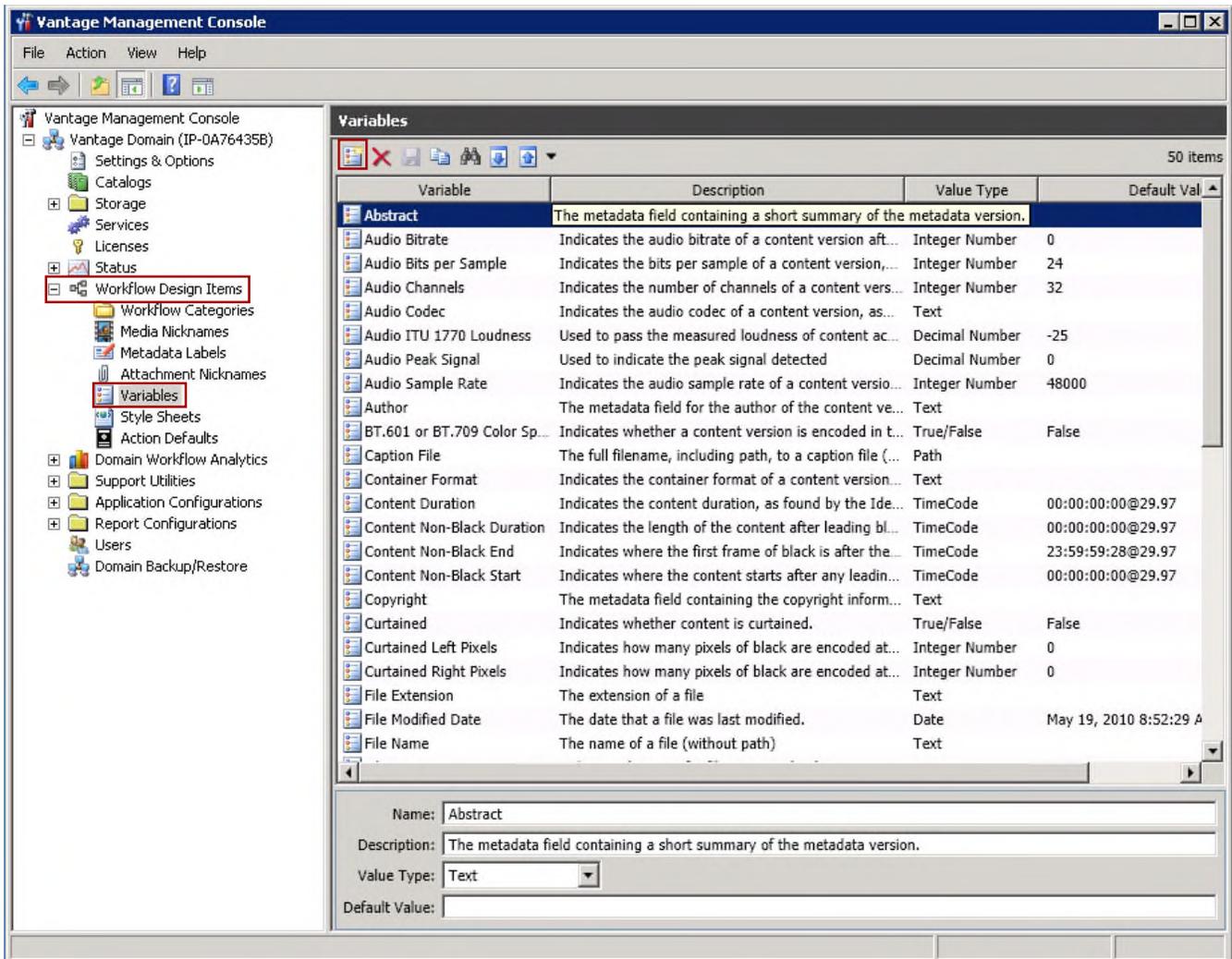
1. Open Vantage Management Console (*Connect to the local machine*).

Figure 163: Vantage Management Console Icon



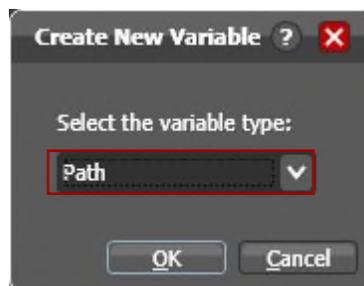
2. Navigate to **Workflow Design Items, Variables, Create New Variable** as shown in the figure on the next page.

Figure 164: Creating a New Variable



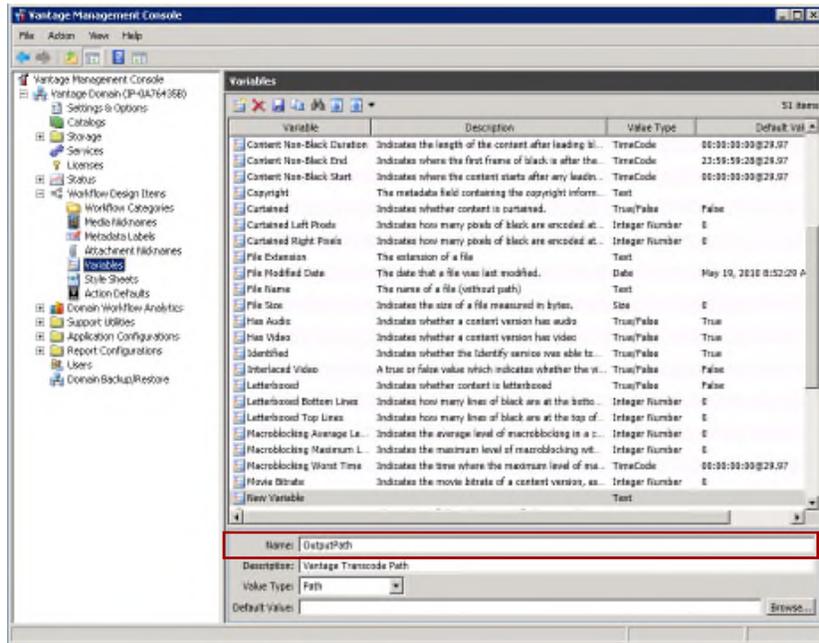
3. Set the Variable Type to Path.

Figure 165: Create New Variable Window



- Update the **Name** to **OutputPath** and click the **Save Icon**.

Figure 166: Updating the Name and Saving the Variable



9.3.2.2 Minimum Vantage Workflow Requirements

- Open Vantage Workflow Designer

Figure 167: Vantage Workflow Designer Icon



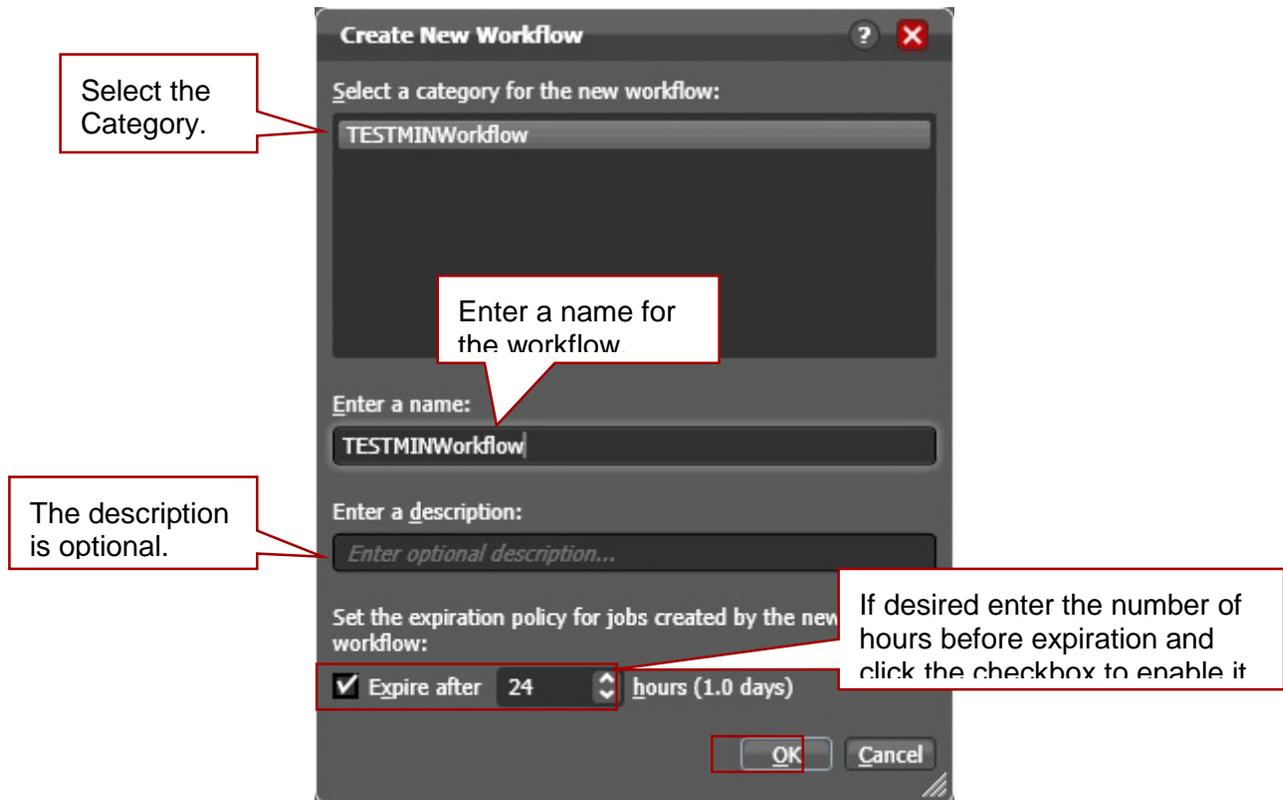
- Create a new **Category**.
 - In the example the Category name is **TESTMINWorkflow**.
 - No spaces or special characters are allowed.**

Figure 168: Create New Category Window



3. Create a **New Workflow** and name it.
 - a. Select the Category for the workflow.
 - v. In the example it is the same name as the Category.
 - b. Entering a Description is optional.
 - c. If desired, select the number of hours for expiration and check the box.
 - d. Click the **OK Button** to save the new workflow.

Figure 169: Create New Workflow Window



4. Click on **Common** and drag down the **Receive Icon**.

Figure 170: Configuring the New Workflow - 1

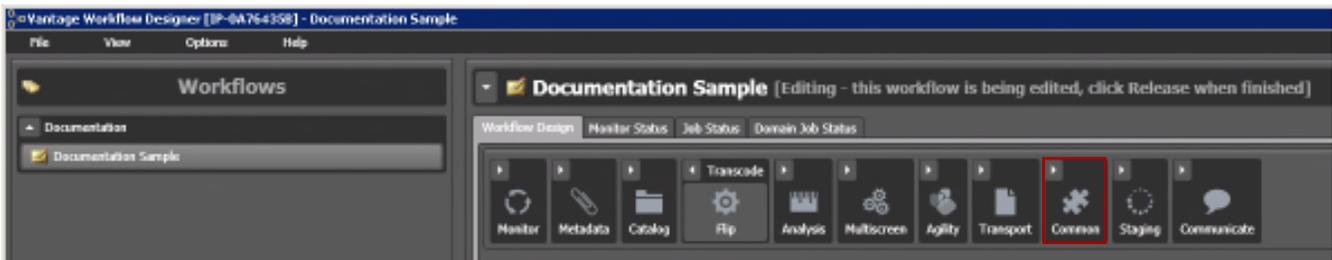
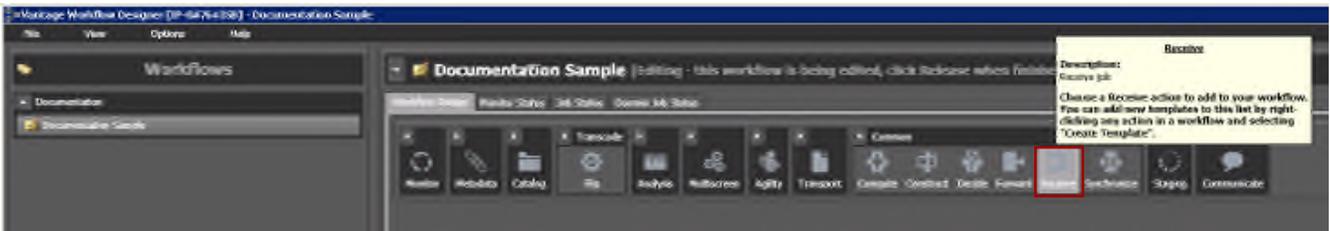


Figure 171: Configuring the New Workflow - 2



5. Click on **Transcode** and drag down the **Flip Icon**.

Figure 172: Configuring the New Workflow - 3

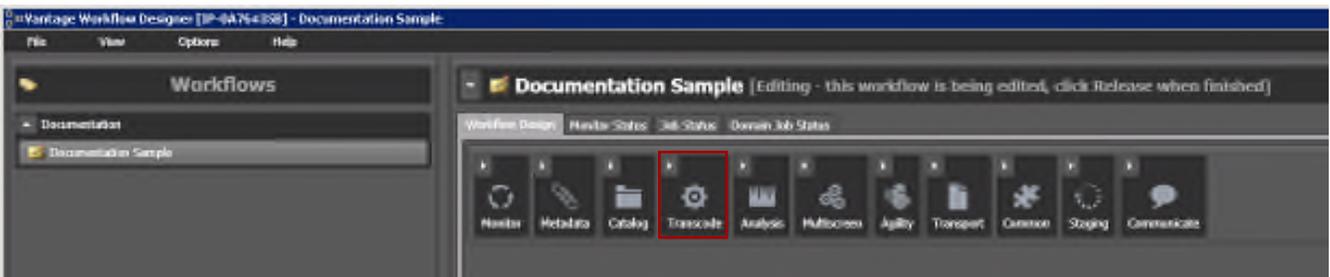
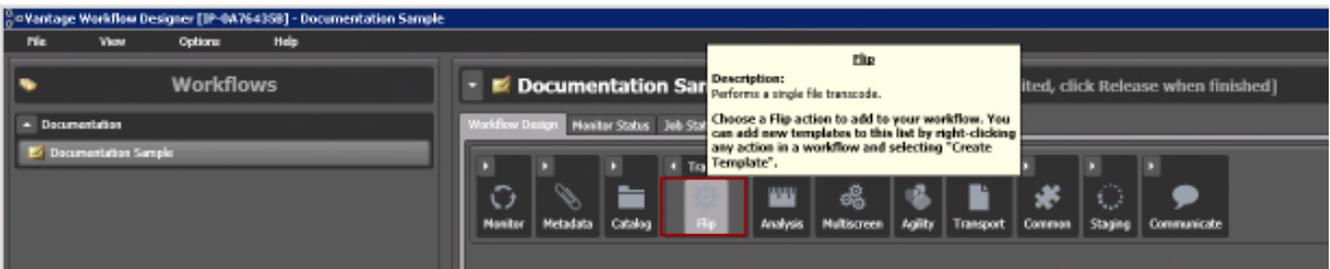
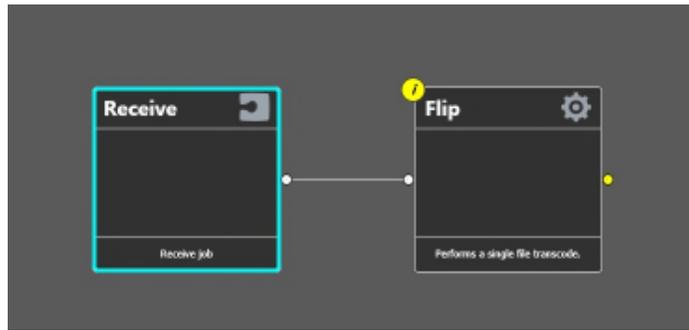


Figure 173: Configuring the New Workflow - 4



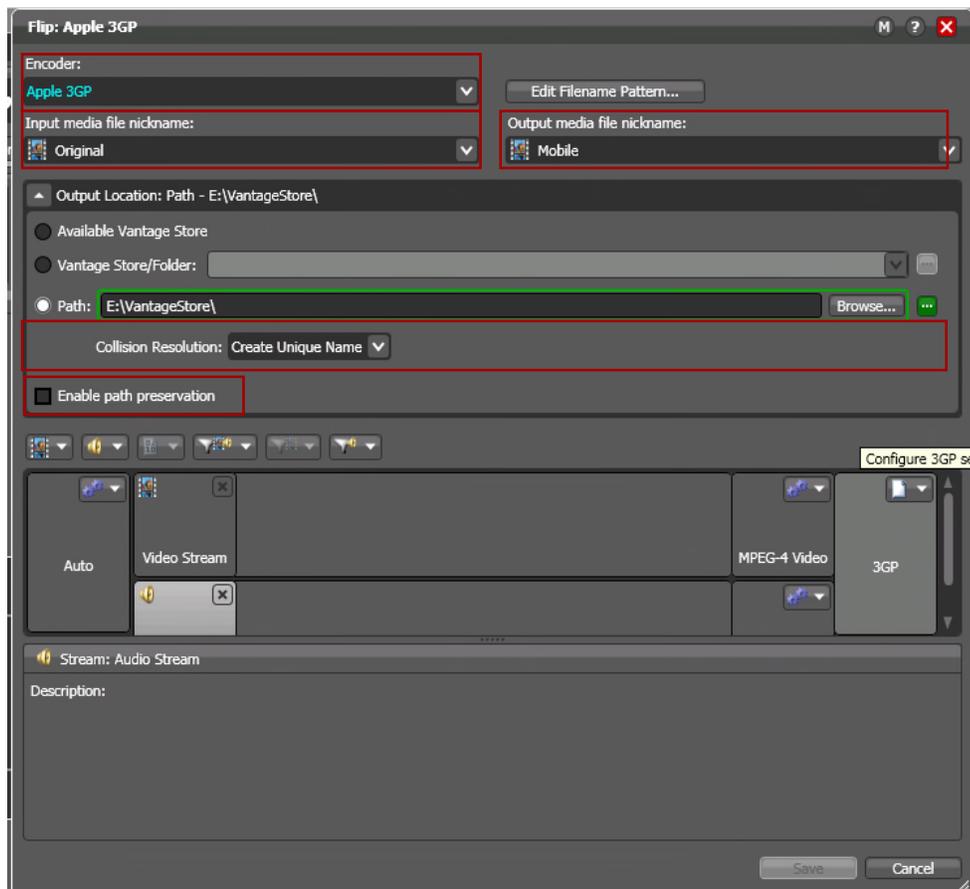
6. Link the **Receive** and **Flip** together.
 - a. Click on the **Receive Yellow Dot** then hold down the mouse button and drag it to the **Flip Yellow Dot**. It should look like the figure below once it is successful.

Figure 174: Linking Receive and Flip Together



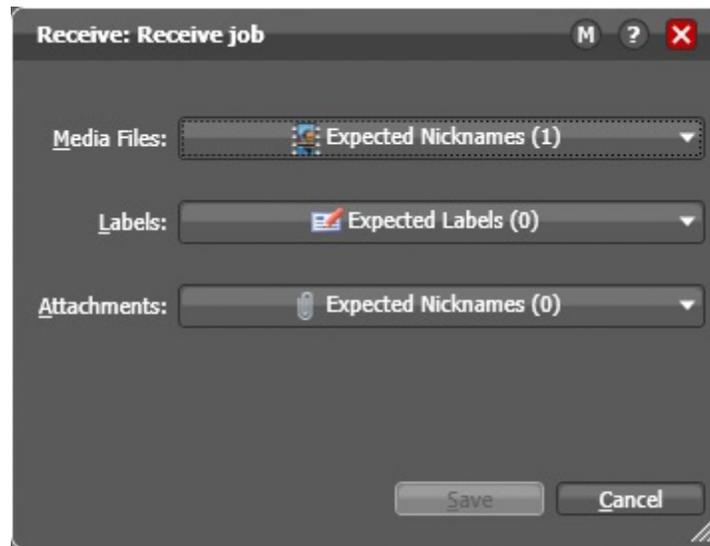
7. Right-click on **Flip** to configure the Flip options.
 - a. In this example a Windows Media file has been configured.
 - vi. **Encoder:** Windows Media
 - vii. **Input media file nickname:** Vantage Proxy
 - viii. **Output media file nickname:** Original
 - b. Expand **Output Location**.
 - c. Choose **Path** then click the **Green “...” Box** and select the variable **OutputPath**.
 - d. Initially set the **Collision Resolution** to **Overwrite**.
 - e. Click the **Save Button** to save the new configuration.

Figure 175: Flip Configuration Window



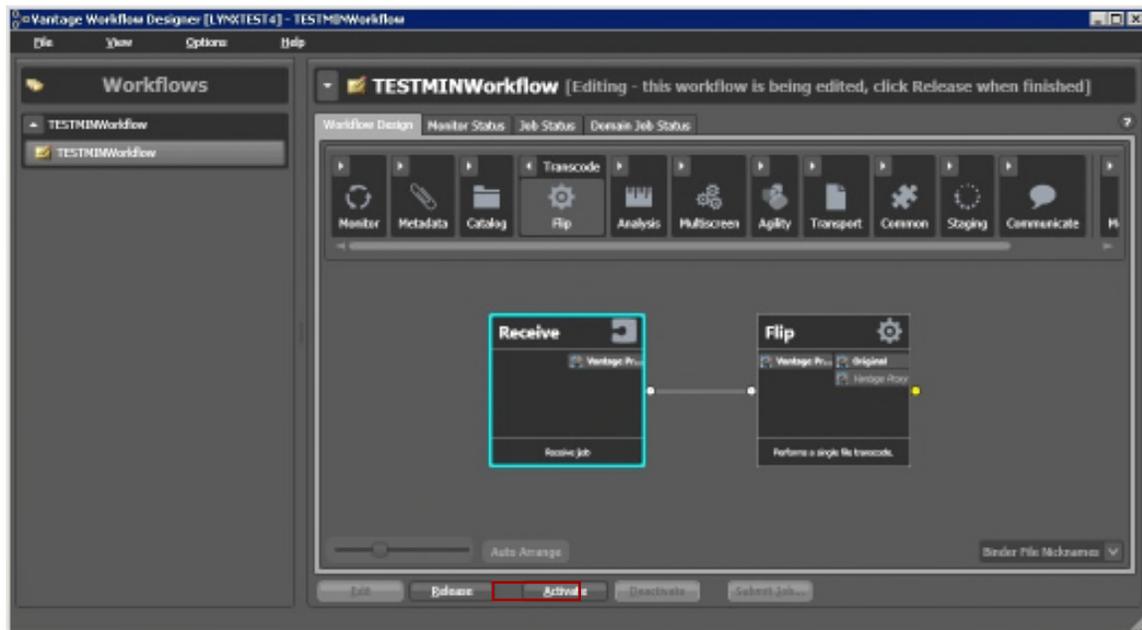
8. Right-click on **Receive** to configure the Receive options.

Figure 176: Receive Configuration Window



9. Click on Media Files and choose Vantage Proxy.
10. Click the **Release Button** so that DIVArchive can use the workflow.

Figure 177: Releasing the Configuration to DIVArchive



9.3.2.3 Vantage Complex Workflow

This Vantage complex workflow example was created for documentation purposes; however it has not been tested with actual media files.

1. Open the Vantage Workflow Designer.

2. Navigate to File and then Create a New Category.
 - a. For this example the Category named **TESTComplex** was created.
3. Navigate to **File** and then **Import Workflow**.
 - a. Browse and select:
 - C:\Program Files
 - (x86)\Telestream\Vantage\Samples\Analysis\Smart SD and HD Transcoding.xml
 - b. Specify the Category created in Step 2 above.
4. At this point it is recommended to change the **Workflow Name** to match the **Category**.
 - a. No space or special characters are allowed.

Figure 178: Vantage Workflow Designer – Renaming the Workflow

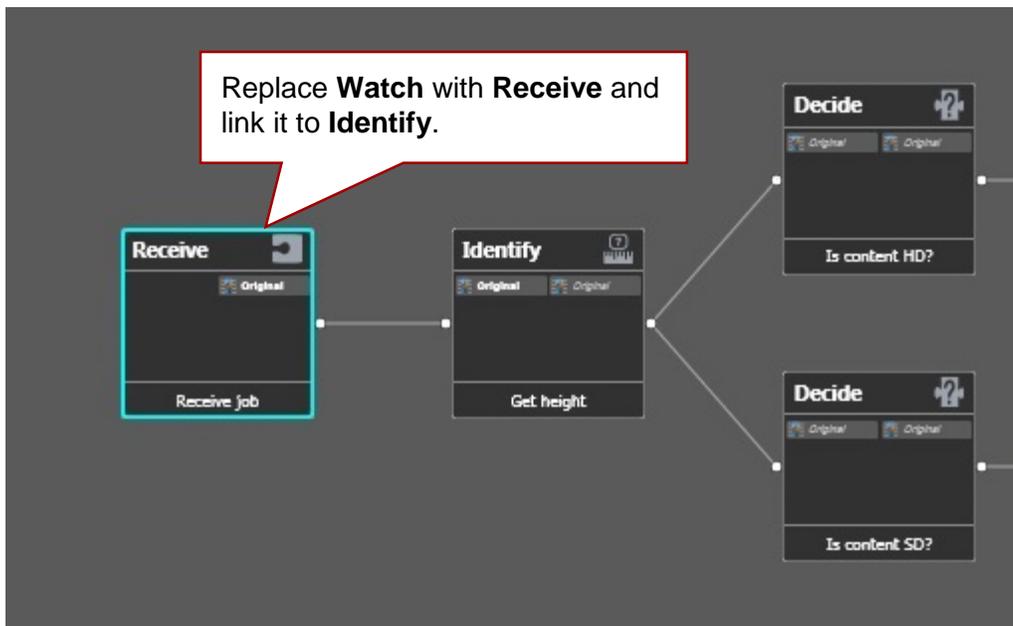


5. Delete the **Watch** and replace it with **Receive**.
 - a. Configure on **Receive** and set **MediaFiles** to **Original**.
 - b. Link this with **Identify**.

Figure 179: Vantage Workflow Designer – Delete Watch

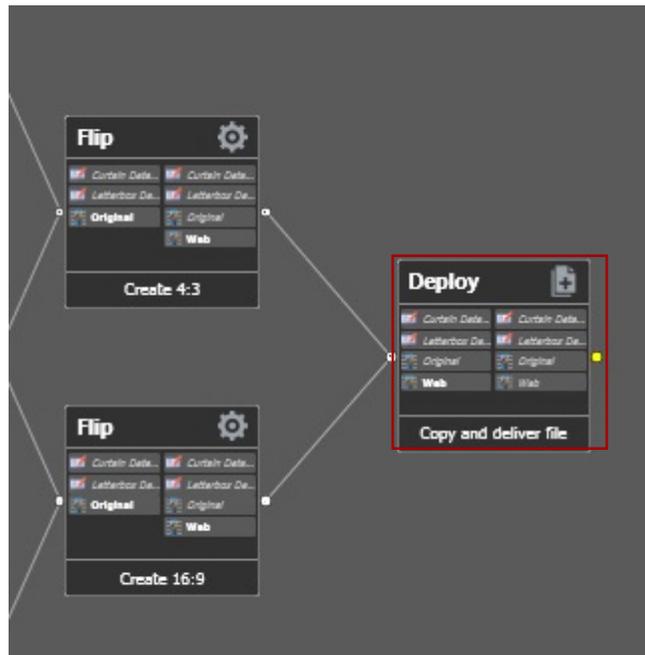


Figure 180: Vantage Workflow Designer – Replace Watch with Receive; Link to Identify



6. Delete **Deploy**.

Figure 181: Deleting Deploy

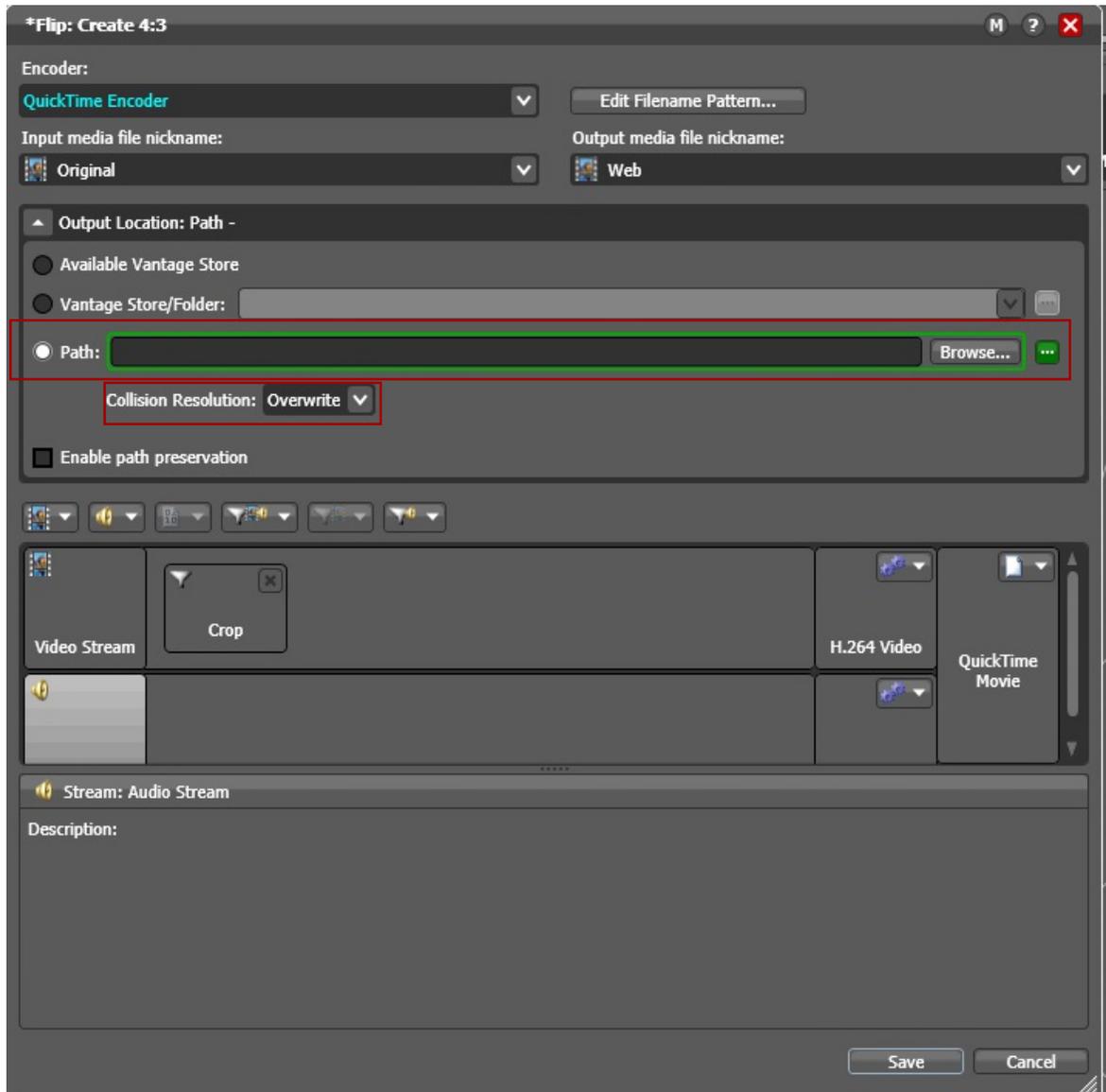


7. Configure both Flip Factories.

- a. Change the **Output Location** to **Path** and then click the **Green "..."** Button and select the **OutputPath**.
- b. Change the **Collision Resolution** to **Overwrite**.

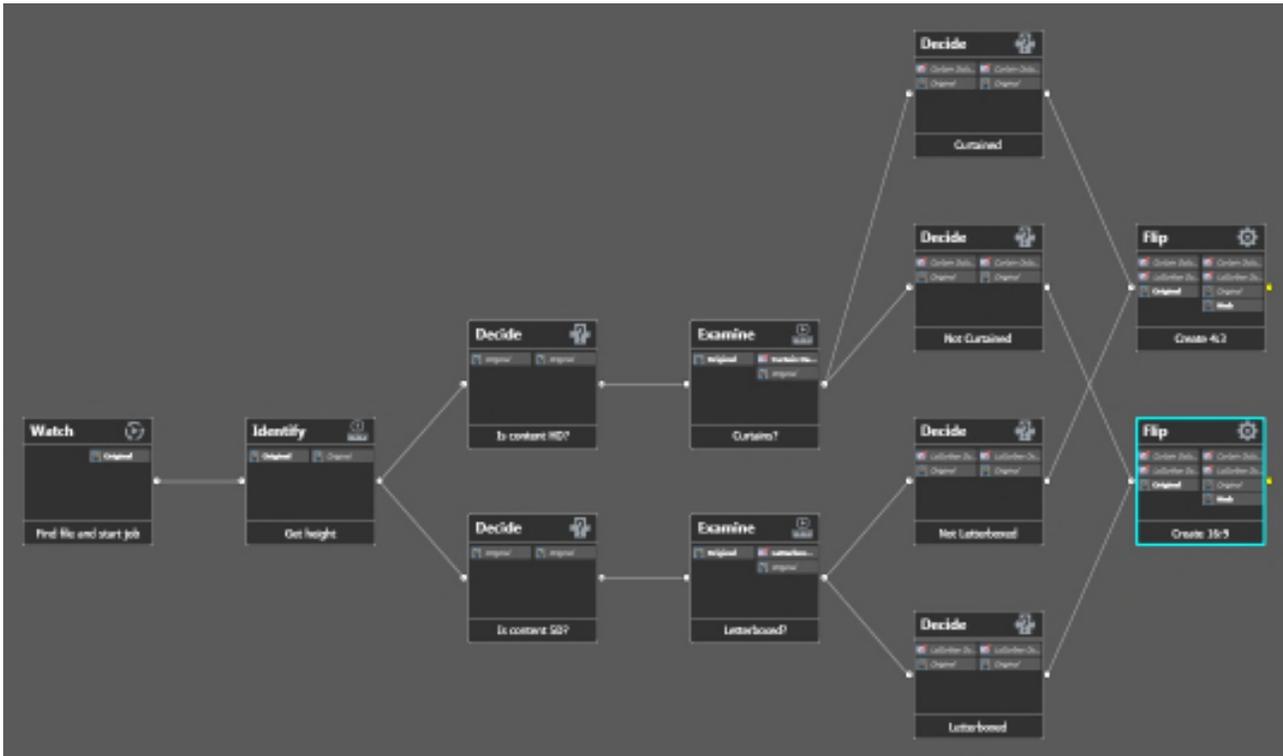
Note: Multiple transcoders are not supported for Flip Factory. They are supported only for Vantage.

Figure 182: Configuring Flip Factory



8. Release the Workflow and it should be ready for DIVArchive. In this example, the workflow should look like the figure below.

Figure 183: Final Complex Workflow Configuration



9.3.3 Setting up Transcoders

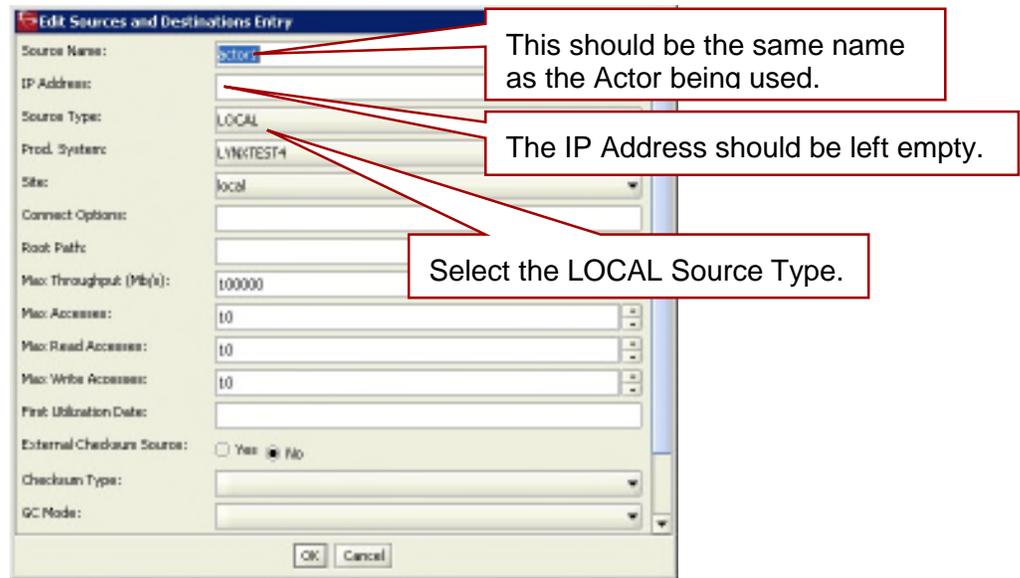
Create a new Vantage transcoder as described in Section 9.3.1.

Set the **Working Directory** to either a local folder or to a path on a remote machine. You can set a remote path for Vantage only. If setting a path to a remote machine, a CIFS UNC path with the appropriate authentication credentials should be specified. The IP address specified in the UNC path must point to the remote machine running the Vantage SDK service.

9.3.4 Setting up Source/Destinations

9. Setup a LOCAL Source/Destination for the Actor.
 - a. **Source Name:** <Same name as the Actor>
 - b. **IP Address:** Leave this empty
 - c. **Source Type:** LOCAL

Figure 184: Configuration Utility Edit Source and Destinations Window - 1



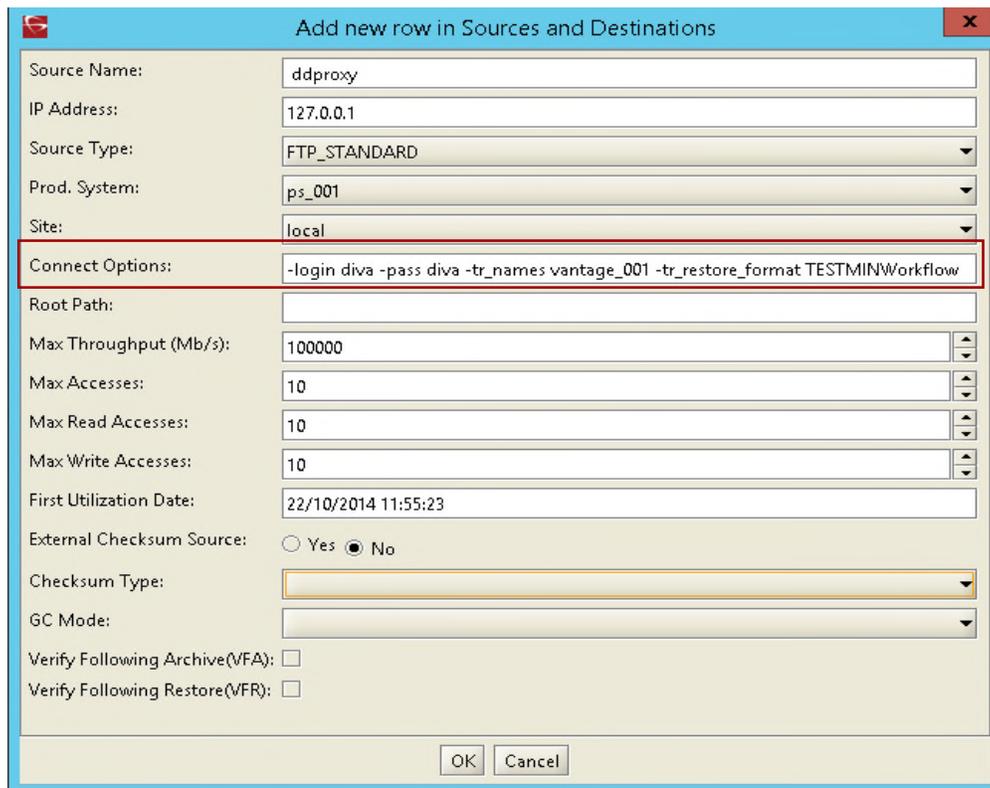
10. Setup the Destination with transcode options.

a. **Vantage Connection Options:**

- i. `-tr_names <TRANSCODER NAME>`
- ii. `-tr_restore_format <WORKFLOW NAME>`

Note: The `auto` format option is valid for Telestream and Bitscream only.

Figure 185: Configuration Utility Edit Source and Destinations Window - 2



10 Movie2Me Integration

Starting with DIVArchive version 7.2.1, Movie2Me Integration is available for use as a special Source/Destination. This Source/Destination uses a separate installation process and requires additional licensing. The integration is compatible with DIVArchive versions 7.2.1 and higher and performs similar to a combination of an FTP Client and FTP Server. This section covers the general configuration and workflow of Movie2Me – operational details can be found in the Movie2Me documentation delivered with your system.

10.1 Minimum Requirements

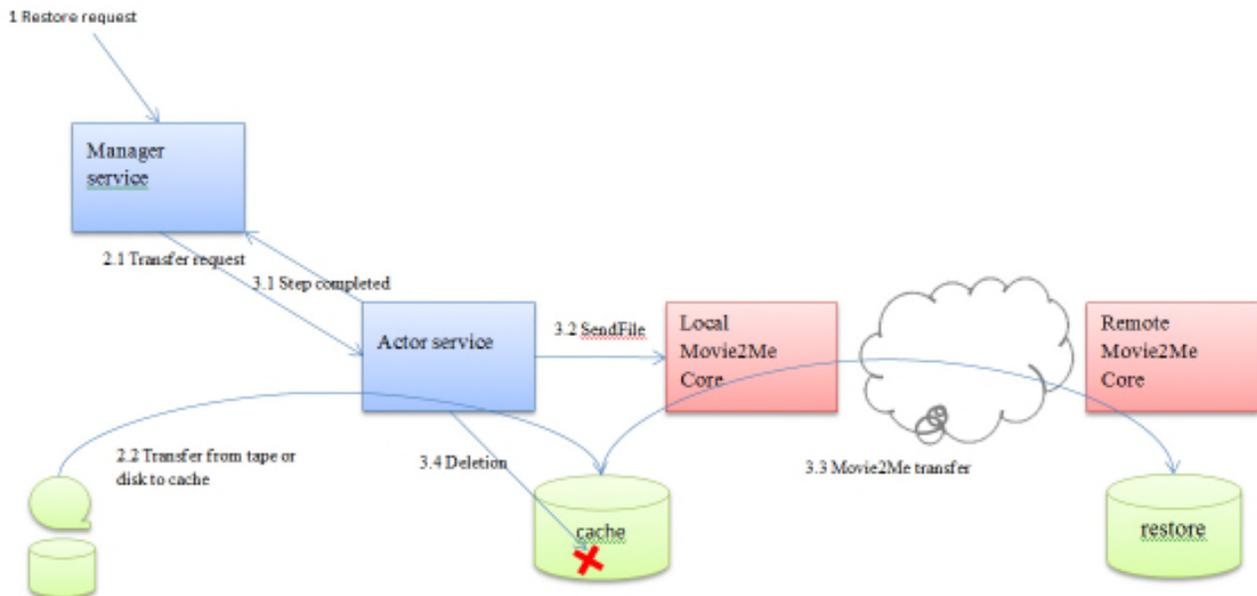
The following are the minimum requirements for a complete Move2Me installation:

- DIVarchive Manager
- DIVArchive Actor
- Local Storage
- Remote Storage
- Local Disk Cache
- Two Movie2Me Cores
 - Local Movie2Me Core (*local to the DIVArchive System*)
 - Remote Movie2Me Core (*remote from the DIVArchive System*)
 - ii. Both Cores have the same software and configuration.
- Cloud between the two cores

10.2 Movie2Me Workflow

The figure and process described below is representative of a basic Movie2Me Workflow. The step numbers refer to the numbering in the diagram.

Figure 186: Movie2Me Basic Workflow



1. A **Restore Request** is received by the Manager.
- 2.1 The Manager sends the transfer request to the selected Actor.
 - a. The purpose of the transfer is to copy the content of the object to the Actor Cache (*with or without AXF encapsulation*).
- 2.2 The Actor performs the transfer from the disk or tape instance to its cache using a Source/Destination.
 - b. The use of a Source/Destination allows more flexibility as some useful options (*-axf, -rm or -rxm1*) may be specified.
 - c. The Actor reports the progress to the Manager as usual.
- a. When the object is in the cache (*transfer completed*), the Manager will receive a **Step Completed Message** from the Actor indicating that the Actor will start the Movie2Me Transfer.
 - d. This allows the Manager to release some resources like the tape and the drive used during the first transfer (*to cache*).
 - e. The Manager will reset the **Request Progress** to 0%.
- 3.2 The Actor connects to a local Movie2Me Core and issues a `sendFile` command for each file to be restored. This could actually be two files in the case of an AXF Restore (*object.axf + object.mdF*).
- 3.3 The Local Movie2Me sends all the files to the Remote Movie2Me. While the transfer is running, the Actor will monitor the progress and report it to the

Manager. When the transfer is 100% complete the Actor will send a **Request Complete** message to the Manager.

- 3.4 The Manager sends a **Server Delete Request** to the Actor.
 - a. The Actor deletes the temporary space in cache.

Notes:

- If **Movie2Me** is used during a restore request to a system's DFM Folder, and the request contains the `-rm` option, the `.mdf` file must be the last file transferred – otherwise an incomplete archive may result. If the `-axf` option is used instead, and the DFM Folder is in Single File Mode, the order of the transfer does not matter.
- The **Movie2Me Core** can be a Restore Folder, Archive Folder, DFM Drop Folder, etc. depending upon the request.

10.3 Movie2Me Configuration

The first part of setting up the Movie2Me Integration requires that a cache disk be configured in DIVArchive and then associated with the Actor where the Movie2Me Local Core will be connecting.

An Array and Disk must be created; only the name of the Array and Disk are relevant. The information below will allow a user to setup the required Array and Disk; however more detailed instructions and screenshots may be found in the DIVArchive Installation and Configuration Guide: Volume I if required.

1. Open the **DIVArchive Configuration Utility** and select the **Drives Tab** and click the **+ Button** in the **Arrays Panel** to add a new Array.
2. Enter the information necessary in the **Array Entry Window** that appears and click the **OK Button** when finished.

Figure 187: DIVArchive Array Entry Window

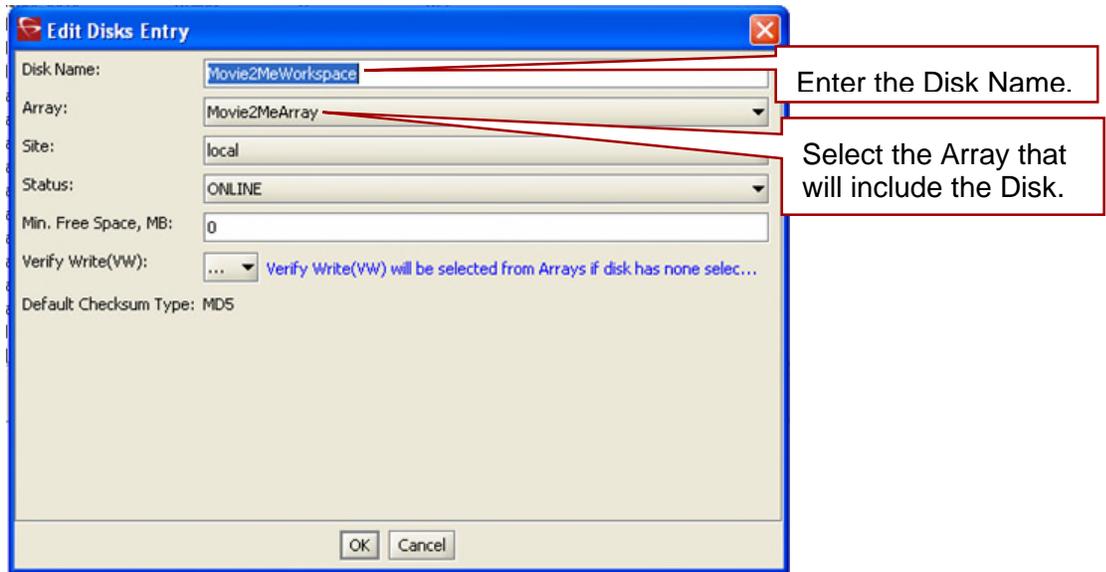
The screenshot shows the 'Edit Arrays Entry' dialog box. The fields are as follows:

ID:	87
Array Name:	Movie2MeArray
Description:	
Format:	AXF
Max. Allowed Disk Space For Repack (%):	100
Verify Write(VW):	OFF
Default Checksum Type:	MDS

Callout boxes provide additional context: one points to the 'Array Name' field, stating 'Enter the Array Name (required) and optional Description (if desired).', and another points to the 'Format', 'Max. Allowed Disk Space For Repack (%)', and 'Verify Write(VW)' fields, stating 'These configuration items are not used.'

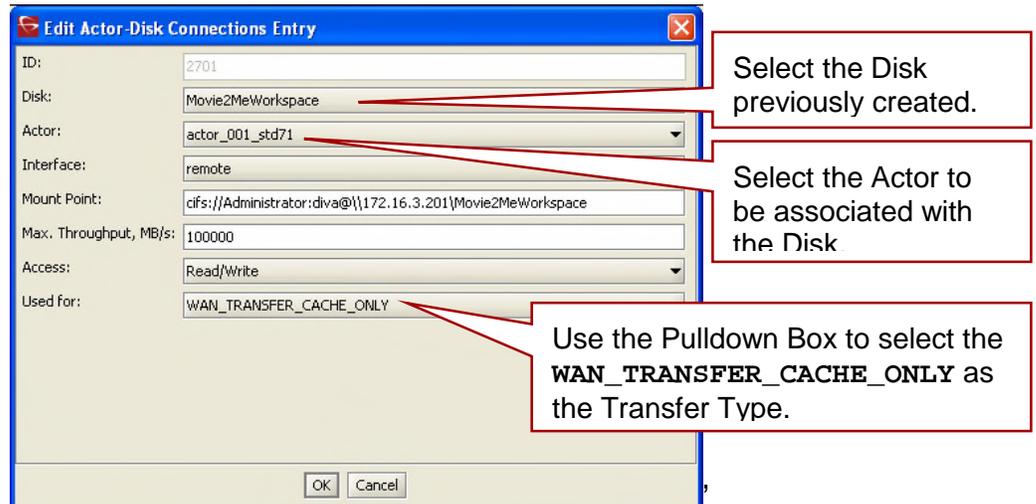
3. In the **DIVArchive Configuration Utility** select the **Disk Tab** and click the **+ Button** in the **Disks Panel**.
 - a. Multiple Disks may be created if there are multiple Movie2Me Cores in the system.
4. Enter the required information in the **Disk Entry Window** that appears and click the **OK Button** when finished.

Figure 188: DIVArchive Disk Entry Window



5. Next the Disk must be associated with an Actor. In the Actor-Disk Connections Window, define the Actors that can see the new disk previously created.
 - a. **This cache must be used for Movie2Me transfers only and therefore the usage must be set to WAN_TRANSFER_CACHE_ONLY.**
 - b. **The mount point is the root directory of the cache from the Actor's point of view.**

Figure 189: DIVarchive Actor-Disk Connection Window



The configuration of the Destination defines how Actors will connect to the Local Movie2Me Core and the Peer ID of the Remote Movie2Me Core. The configuration requires the following options (*also shown in the figure below*):

- **-login** and **-pass**
 - This is the login/password of the Local Movie2Me Core HTTP Server.
- The IP Address of the local Movie2Me Core.
- **-port** is the listening port of the Local Movie2Me Core HTTP Server. In the example below, the Actor will use the parameter to be able to connect to <http://172.16.3.201:8080/transferService>.
- **-peer** is the ID of the Remote Movie2Me Core; however, there is a way to obtain the ID from the Local Movie2Me Core:
 1. Stop the **m2m-core Service**.
 2. Open the file `c:\ProgramData\M2M-Core\database\m2mdb.script`.
 3. Look for entries similar to: `INSERT INTO PEER VALUES(`.
 4. The first value in braces will be the **Peer-ID**.

Figure 190: DIVarchive Sources and Destination Entry Window

The screenshot shows a window titled "Edit Sources and Destinations Entry". The fields are as follows:

Source Name:	m2mBaltika
IP Address:	172.16.3.201
Source Type:	MOVIE2ME
Prod. System:	sony_smss
Site:	local
Connect Options:	-login diva -pass diva -peer 2
Root Path:	
Max Throughput (Mb/s):	100000
Max Accesses:	10
Max Read Accesses:	10
Max Write Accesses:	10
First Utilization Date:	
External Checksum Source:	<input type="radio"/> Yes <input checked="" type="radio"/> No
Checksum Type:	
GC Mode:	
Verify Following Archive(VFA):	<input type="checkbox"/>
Verify Following Restore(VFR):	<input type="checkbox"/>

Buttons: OK, Cancel

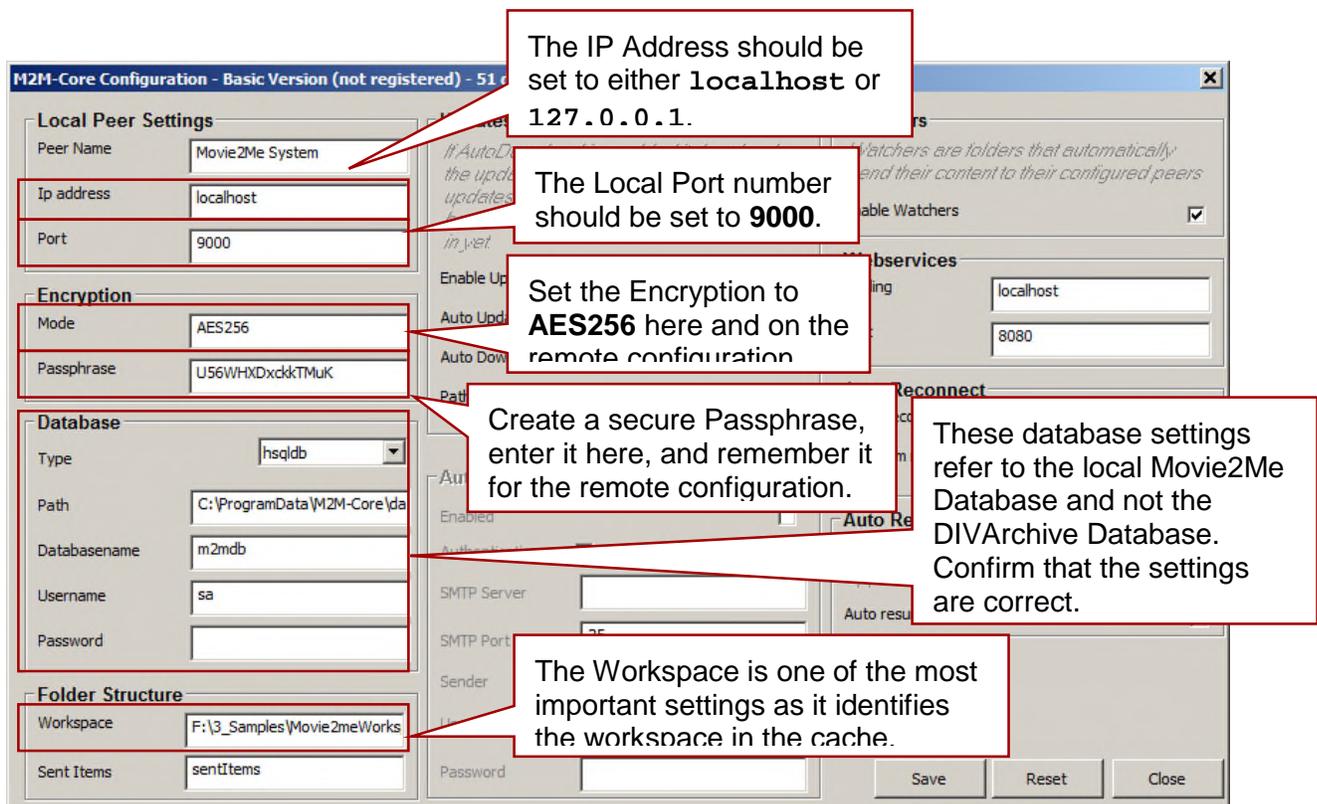
NOTES:

- Archives are not supported from this type of Source/Destination.
- N-Restore is not supported.
- The Cache Type can be Remote or Local only (*not FTP, Mediagrid or Metasan*).
- Delete and Write QOS is not supported.

- Do Not Overwrite QOS is not supported.
- VFR is not supported.
- This configuration is for a single Movie2Me Core per DIVArchive Production System.
- Single Actor-Disk Connection with WAN_TRANSFER option per Actor. In other words, one Actor cannot see more than one WAN_TRANSFER cache.

Once installed, Movie2Me uses its own utility for configuration as described in the figures below. The first figure is the configuration screen and entries for the Movie2Me Local Core and the second figure is for the Movie2Me Remote Core. The configuration should be the same on both cores as indicated.

Figure 191: Local Movie2Me Core Configuration



Note: The Workspace is the mount point identified in the DIVArchive configuration for the system's cache. The mount point is created by setting up a disk in DIVArchive and associating that disk with the selected Actor and should be set to use the WAN_Transfer_Cache_Only Usage Type (*new Usage Type*).

Figure 192: Remote Movie2Me Core Configuration

Edit Peer - Basic Version (not registered) - 51 days left

General

Name:

IP:

Priority: (0 = highest)

Encryption

Mode: AES no encryption

Passphrase:

Login

Name:

Password:

Encryption Callout: The encryption should be set the same as in the Local Movie2Me Configuration. Confirm that the AES Encryption type is used and that the PassPhrase matches the entries on the local configuration

Connection

Port: (1 - 65535)

Connection Callout: The Port Number must be the same as on the Local Movie2Me Configuration.

Max Bandwidth (decimal marker: '.'; -1 = unlimited): Mbit/s

Max number of uploads: Max. number of downloads:

APPENDIX A – Dynamic Configuration Changes

A1 Introduction

This manual lists the currently supported changes to your configuration that can be made effective while the Manager is running, and those that require a software component and/or the DIVArchive Manager to be restarted.

A2 DIVArchive Manager Configuration Updates

If a parameter in the DIVArchive Manager Configuration File is changed, the following table lists what is currently required for that change to take effect. **Manager Reload** indicates the **reload** option should be used from a command line/terminal window.

Table 41: DIVArchive Manager Configuration Updates

Parameter Change	Effective After
<code>SERVICE_NAME</code>	Reinstall and restart of service
<code>DIVAMANAGER_NAME</code>	Manager restart
<code>DIVAMANAGER_PORT</code>	Manager restart
<code>DIVAMANAGER_TNSNAME</code>	Manager restart
<code>DIVAMANAGER_DBHOST</code>	Manager restart
<code>DIVAMANAGER_DBPORT</code>	Manager restart
<code>DIVAMANAGER_DBSID</code>	Manager restart
<code>DIVAMANAGER_DBUSER</code>	Manager restart
<code>DIVAMANAGER_DBPASSWORD</code>	Manager restart
<code>DIVAMANAGER_TO_LOWER</code>	Manager reload
<code>DIVAMANAGER_MAX_CONNECTIONS</code>	Manager restart
<code>DIVAMANAGER_MAX_SIMULTANEOUS_REQUESTS</code>	Manager reload
<code>DIVAMANAGER_MAX_INACTIVE_REQUESTS</code>	Manager reload
<code>DIVAMANAGER_TYPICAL_OBJECT_SIZE</code>	Manager restart
<code>DIVAMANAGER_MAX_SPAN_SEGMENTS</code>	Manager reload
<code>DIVAMANAGER_CAPACITY_LOW_WATER_MARK</code>	Manager restart

Parameter Change	Effective After
DIVAMANAGER_MAX_OBJECTS_FOR_REPACK	Manager reload
DIVAMANAGER_STOP_IMMEDIATELY_FOR_REPACK	Manager restart
DIVAMANAGER_TIME_TO_WAIT_FOR_GRACEFUL_SHUTDOWN	Manager restart
DIVAMANAGER_DISMOUNT_AFTER	Manager restart
DIVAMANAGER_UPDATE_PRIORITIES_PERIOD	Manager restart
DIVAMANAGER_MAX_DELAY_BETWEEN_SCHEDULER	Manager reload
DIVAMANAGER_SCHEDULER_AFTER_INACTIVITY	Manager reload
DIVAMANAGER_PING_INTERVAL	Manager restart
DIVAMANAGER_EXPORT_ROOT_DIR	Manager reload
DIVAMANAGER_MAX_RESTORE_SERVERS	Manager reload
DIVAMANAGER_MAX_EXPORT_TAPES	Manager reload
DIVAMANAGER_MAX_EXPORT_ELEMENTS	Manager reload
DIVAMANAGER_MAX_FILES_IN_ARCHIVE	Manager reload
DIVAMANAGER_MAX_FILES_IN_PARTIAL_RESTORE	Manager reload
USE_IMPROVED_BEST_WORST_FIT_ALGORITHM	Manager reload
DIVAMANAGER_SITE_SUPPORT_ENABLED	Manager reload
DIVAMANAGER_CACHE_QOS_USE_DISK	Manager reload
DIVAMANAGER_PRIORITY_TIER	Manager reload
DIVAMANAGER_ETC_FEATURE	Manager restart
DIVAMANAGER_ETC_CONFIDENCE_LEVEL	Manager restart
DIVAMANAGER_OVERWRITE_POLICY	Manager reload
DIVAMANAGER_OVERWRITE_OVERRIDE	Manager reload
DIVAMANAGER_BULK_EXPORT	Manager reload
ATTEMPT_ACCESS_TO_OFFLINE_DISK	Manager reload
CHANGE_DISK_STATE_ON_ERROR	Manager reload

Parameter Change	Effective After
MANAGER_ACTOR_DISK_RETRY_NUMBER	Manager reload
DISK_STATUS_POLLING_RATE	Manager reload
DISK_BUFFER_SPACE	Manager reload
DISK_CONNECTION_STATE_RESET_DELAY	Manager reload
DIVAMANAGER_MAX_EXCLUDED_INSTANCES	Manager reload
DIVAMANAGER_REQUEST_SCHEDULING_QUEUE_SIZE	Manager reload
DIVAMANAGER_API_TASK_QUEUE_SIZE	Manager reload
DIVAMANAGER_MAX_CONCURRENT_REQUESTS	Manager reload
DIVAMANAGER_MIN_DB_CONNECTION_LIMIT	Manager reload
DIVAMANAGER_MAX_DB_CONNECTION_LIMIT	Manager reload
DIVAMANAGER_INITIAL_DB_CONNECTION_LIMIT	Manager reload
DIVAMANAGER_INACTIVITY_TIMEOUT	Manager reload
DIVAMANAGER_SIZE_OF_STATEMENT_CACHE	Manager reload
DIVAMANAGER_DEFAULT_ROW_PREFETCH	Manager reload
DIVAMANAGER_FAILOVER_ENABLED	Manager reload
DIVAMANAGER_NUM_RS_SOLUTIONS_TO_EVALUATE	Manager reload

A3 Configuration Utility Changes in the Systems Tab

A3.1 Production Systems Frame

If a parameter in the **Production Systems** frame of the **Systems** tab in the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 42: Production Systems Frame Parameter Changes

Parameter Change	Effective After
Add New	Notify Manager
Delete	Notify Manager
Production System Name	Notify Manager

A3.2 Sites Frame

If a parameter in the **Sites** frame of the **Systems** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 43: Sites Frame Parameter Changes

Parameter Change	Effective After
Add New Site	Notify Manager
Delete a Site	Notify Manager
Site Name	Notify Manager
Is Main Site	Notify Manager
Comments	Notify Manager

A3.3 Sources and Destinations Frame

If a parameter in the **Sources and Destinations** frame of the **Systems** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 44: Sources and Destinations Frame Parameter Changes

Parameter Change	Effective After
Add New	Notify Manager
Delete	Notify Manager, effective after requests completed.
Source Name	Notify Manager, effective after requests completed.
IP Address	Notify Manager, effective after requests completed.
Source Type	Notify Manager, effective after requests completed.
Production System	Notify Manager, effective after requests completed.
Site	Notify Manager, effective after requests completed.
Connect Option	Notify Manager, effective after requests completed.
Root Path	Notify Manager, effective after requests completed.
Max Throughput	Notify Manager, effective after requests completed.
Max Accesses	Notify Manager, effective after requests completed. It is recommended that no changes to this parameter be made while there are active requests; it can lead to request abortion.
Max Read Accesses	Notify Manager, effective after requests completed. It is recommended that no changes to this parameter be made while there are active requests; it can lead to request abortion.
Max Write Accesses	Notify Manager, effective after requests completed. It is recommended that no changes to this parameter be made while there are active requests; it can lead to request abortion.

A3.4 Actors Frame

If a parameter in the **Actors** frame of the **Systems** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 45: Actors Frame Parameter Changes

Parameter Change	Effective After
Add New Actor	Notify Manager
Delete Actor	Notify Manager (<i>must disconnect Actor first</i>).
Actor Name	Notify Manager (<i>must disconnect Actor first</i>).
IP Address	Notify Manager (<i>must disconnect Actor first</i>).
Port	Notify Manager (<i>must disconnect Actor first</i>).
Production System	Notify Manager, effective after requests completed
Site	Notify Manager, effective after requests completed
Max Drive Operations	Notify Manager, effective after requests completed
Max Server Operations	Notify Manager, effective after requests completed
Max Disk Operations	Notify Manager, effective after requests completed
Direct Restore	Notify Manager, effective after requests completed
Cache Restore	Notify Manager, effective after requests completed
Copy to Group	Notify Manager, effective after requests completed
Associative Copy	Notify Manager, effective after requests completed
Repack	Notify Manager, effective after requests completed
Delete	Notify Manager, effective after requests completed
Direct Archive	Notify Manager, effective after requests completed
Cache Archive	Notify Manager, effective after requests completed

A3.5 Transcoders Frame

If a parameter in the **Transcoders** frame of the **Systems** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 46: Transcoders Frame Parameter Changes

Parameter Change	Effective After
Add New	Notify Manager
Delete	Notify Manager
Transcoder Name	Notify Manager
Transcoder Type	Notify Manager
Transcoder Port	Notify Manager
Working Directory	Notify Manager
Executable Path	Notify Manager
Performance	Notify Manager

A4 Configuration Utility Changes in the Robots Tab

A4.1 Robot Managers Frame

If a parameter in the **Robot Managers** frame of the **Robots** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 47: Robot Managers Frame Parameter Changes

Parameter Change	Effective After
Add New	Notify Manager
Delete	Notify Manager
Robot Manager Name	Notify Manager
Address	Notify Manager (<i>must disconnect RM first</i>)
Port	Notify Manager (<i>must disconnect RM first</i>)
Site	Notify Manager

A4.2 Media Compatibility Frame

If a parameter in the **Media Compatibility** frame of the **Robots** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 48: Media Compatibility Frame Parameter Changes

Parameter Change	Effective After
Delete	Notify Manager

A4.3 Robot Managers – ACS Frame

If a parameter in the **Robot Managers – ACS** frame of the **Robots** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 49: Robot Managers – ACS Frame Parameter Changes

Parameter Change	Effective After
Delete	Notify Manager

A5 Configuration Utility Changes in the Disks Tab

A5.1 Arrays Frame

If a parameter in the **Arrays** frame of the **Disks** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 50: Arrays Frame Parameter Changes

Parameter Change	Effective After
Add New	Notify Manager
Delete	Notify Manager
Array Name	Notify Manager
Description	Notify Manager

A5.2 Disks Frame

If a parameter in the **Disks** frame of the **Disks** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 51: Disks Frame Parameter Changes

Parameter Change	Effective After
Add New	Notify Manager
Delete	Notify Manager
Disk Name	Notify Manager
Array	Notify Manager
Site	Notify Manager
Status	Notify Manager
Min Free Space	Notify Manager

A5.3 Actor-Disk Connections Frame

If a parameter in the **Actors –Disk Connections** frame of the **Disks** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 52: Actor-Disk Connections Frame Parameter Changes

Parameter Change	Effective After
Add New	Notify Manager
Delete	Notify Manager
Disk	Notify Manager
Actor	Notify Manager
Interface	Notify Manager
Mount Point	Notify Manager
Max Throughput	Notify Manager
Access	Notify Manager
Used For	Notify Manager

A6 Configuration Utility Changes in the Drives Tab

A6.1 Drives Frame

If a parameter in the **Drives** frame of the **Drives** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 53: Drives Frame Parameter Changes

Parameter Change	Effective After
Delete	Notify Manager
Serial Number	Notify Manager
Status	Notify Manager
Enabled Operations	Notify Manager
Used	Manager Restart
Installation Date	Effective immediately
Last Upgrade Date	Effective immediately
Last Cleaning Date	Effective immediately

A6.2 Libraries Frame

If a parameter in the **Libraries** frame of the **Drives** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 54: Libraries Frame Parameter Changes

Parameter Change	Effective After
Delete	Notify Manager
Name	Notify Manager
Serial Number	Notify Manager
Status	Notify Manager

A6.3 Drive Properties Frame

If a parameter in the **Drive Properties** frame of the **Drives** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 55: Drive Properties Frame Parameter Changes

Parameter Change	Effective After
Add (via Sync DB)	Notify Manager
Delete	Notify Manager

A6.4 Actor – Drives Frame

If a parameter in the **Actor-Drives** frame of the **Drives** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 56: Actor – Drives Frame Parameter Changes

Parameter Change	Effective After
Add New	Notify Manager
Delete	Notify Manager
Actor	Notify Manager
Drive	Notify Manager

A7 Configuration Utility Changes in the Tapes Tab

If a parameter in the following frames of the **Tapes** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 57: Configuration Utility Changes in the Tapes Tab

Parameter Change	Effective After
Tape Properties	Notify Manager
Empty Ejected Tapes	Effective immediately
Inserted Protected Tapes	Effective immediately
Tape States	Effective immediately

A8 Configuration Utility Changes in Sets, Groups and Mapping Tab

Changes made in this section become effective as soon as they are applied. No manual update is necessary.

A9 Configuration Utility Changes in the DIVAprotect Tab

If a parameter in the following frames of the **DIVAprotect** tab of the Configuration Utility is changed, the following table lists what is currently required for that change to take effect.

Table 58: Configuration Utility Changes in the DIVAprotect Tab

Parameter Change	Effective After
Configuration	Notify Manager
Event Definitions	Currently cannot be altered
Metric Definitions	Effective immediately

A10 Configuration Utility Changes in the Storage Plans Tab

Changes made in this tab are effective immediately. **It is highly recommended that the Storage Plan Manager Service be stopped prior to altering any setting in this tab.**

A11 Configuration Utility Changes in the Slots Tab

Changes made in this tab are effective immediately. **It is highly recommended that the Storage Plan Manager Service be stopped prior to altering any setting in this tab.**

Table 60: Metric Definitions

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
TAPE_DRIVE_DATA_RATE	Tape Drive: data rate	TAPE READ TAPE WRITE	Drive Serial Number	Transfer Rate	Average	1 day	Yes
TAPE_DRIVE_ERROR_RATE	Tape Drive: internal error rate	TAPE READ TAPE WRITE	Drive Serial Number	Transfer Error Rate	Average	1 day	Yes
TAPE_DRIVE_TIME_ALL_OPERATION_DAY	Tape Drive: time in ALL operation	TAPE INSERT TAPE MOUNT TAPE POSITION TAPE READ TAPE WRITE TAPE UNLOAD TAPE DISMOUNT TAPE EJECT	Drive Serial Number	Event Duration	SUM	1 day	Yes
TAPE_DRIVE_TIME_ALL_OPERATION	Tape Drive: time in ALL operation	TAPE INSERT TAPE MOUNT TAPE POSITION TAPE READ TAPE WRITE TAPE UNLOAD TAPE DISMOUNT TAPE EJECT	Drive Serial Number	Event Duration	SUM	Lifetime	Yes
TAPE_DRIVE_TIME_READ_DAY	Tape Drive: time in READ operation	TAPE READ	Drive Serial Number	Event Duration	SUM	1 day	Yes
TAPE_DRIVE_TIME_READ	Tape Drive: time in READ operation	TAPE READ	Drive Serial Number	Event Duration	SUM	Lifetime	Yes
TAPE_DRIVE_TIME_WRITE_DAY	Tape Drive: time in WRITE operation	TAPE WRITE	Drive Serial Number	Event Duration	SUM	1 day	Yes
TAPE_DRIVE_TIME_WRITE	Tape Drive: time in WRITE operation	TAPE WRITE	Drive Serial Number	Event Duration	SUM	Lifetime	Yes
TAPE_DRIVE_LAST_OPERATION_DATE	Tape Drive: date of last MOUNT, DISMOUNT, READ or WRITE	TAPE MOUNT TAPE READ TAPE WRITE TAPE DISMOUNT	Drive Serial Number	Event End Time	MAX	Lifetime	Yes
TAPE_DRIVE_NUMBER_MOUNTS	Tape Drive: number of mounts	TAPE MOUNT	Drive Serial Number	Drive Serial Number	Count	Lifetime	Yes
TAPE_DRIVE_READ_WRITE_NUMBER	Tape Drive: number of READ and WRITE operations (together)	TAPE READ TAPE WRITE	Drive Serial Number	Drive Serial Number	Count	Lifetime	Yes
TAPE_DRIVE_READ_WRITE_NUMBER_DAY	Tape Drive: number of READ and WRITE operations (together)	TAPE READ TAPE WRITE	Drive Serial Number	Drive Serial Number	Count	1 day	Yes
TAPE_DRIVE_READ_WRITE	Tape Drive: amount of data READ and WRITTEN (together)	TAPE READ TAPE WRITE	Drive Serial Number	Transfer Size	SUM	Lifetime	Yes
TAPE_DRIVE_READ_WRITE_DAY	Tape Drive: amount of data READ and WRITTEN (together)	TAPE READ TAPE WRITE	Drive Serial Number	Transfer Size	SUM	1 day	Yes
TAPE_DRIVE_NUMBER_READ_WRITE_ABORTED	Tape Drive: number of aborted READ and WRITE operations (together)	TAPE READ ERR TAPE WRITE ERR	Drive Serial Number	Drive Serial Number	Count	Lifetime	Yes
TAPE_DRIVE_NUMBER_READ_WRITE_ABORTED_DAY	Tape Drive: number of aborted READ and WRITE operations (together)	TAPE READ ERR TAPE WRITE ERR	Drive Serial Number	Drive Serial Number	Count	1 day	Yes

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
TAPE_DRIVE_NUMBER_MOUNT_DISMOUNT_ABORTED	Tape Drive: number of aborted MOUNT and DISMOUNT operations (together)	TAPE MOUNT ERR TAPE DISMOUNT ERR	Drive Serial Number	Drive Serial Number	Count	Lifetime	Yes
TAPE_DRIVE_OPERATION_TOTAL_TIME_DAY	Tape Drive: total time of drive operation	TAPE READ TAPE WRITE	Drive Serial Number	Event Duration	SUM	1 day	Yes
TAPE_DRIVE_OPERATION_TOTAL_TIME	Tape Drive: total time of drive operation	TAPE READ TAPE WRITE	Drive Serial Number	Event Duration	SUM	Lifetime	Yes
TAPE_LIBRARY_NUMBER_MOUNT_DAY	Tape Library : total number of MOUNT operation	TAPE MOUNT	Library Serial Number	Event Id	Count	1 day	Yes
TAPE_LIBRARY_NUMBER_MOUNT	Tape Library : total number of MOUNT operation	TAPE MOUNT	Library Serial Number	Event Id	Count	Lifetime	Yes
TAPE_LIBRARY_NUMBER_MOUNT_ABORTED_DAY	Tape Library : total number of ABORTED MOUNT operation	TAPE MOUNT ERR	Library Serial Number	Event Id	Count	1 day	Yes
TAPE_LIBRARY_NUMBER_MOUNT_ABORTED	Tape Library : total number of ABORTED MOUNT operation	TAPE MOUNT ERR	Library Serial Number	Event Id	Count	Lifetime	Yes
TAPE_LIBRARY_NUMBER_DISMOUNT_ABORTED_DAY	Tape Library : total number of ABORTED DISMOUNT operation	TAPE DISMOUNT ERR	Library Serial Number	Event Id	Count	1 day	Yes
TAPE_LIBRARY_NUMBER_DISMOUNT_ABORTED	Tape Library : total number of ABORTED DISMOUNT operation	TAPE DISMOUNT ERR	Library Serial Number	Event Id	Count	Lifetime	Yes
TAPE_LIBRARY_NUMBER_READ_DAY	Tape Library : total number of READ operation	TAPE READ TAPE READ ERR	Library Serial Number	Event Id	Count	1 day	Yes
TAPE_LIBRARY_NUMBER_READ	Tape Library : total number of READ operation	TAPE READ TAPE READ ERR	Library Serial Number	Event Id	Count	Lifetime	Yes
TAPE_LIBRARY_NUMBER_WRITE_DAY	Tape Library : total number of WRITE operation	TAPE WRITE TAPE WRITE ERR	Library Serial Number	Event Id	Count	1 day	Yes
TAPE_LIBRARY_NUMBER_WRITE	Tape Library : total number of WRITE operation	TAPE WRITE TAPE WRITE ERR	Library Serial Number	Event Id	Count	Lifetime	Yes
TAPE_LIBRARY_READ_DAY	Tape Library : total amount of data READ	TAPE READ	Library Serial Number	Transfer Size	SUM	1 day	Yes
TAPE_LIBRARY_READ	Tape Library : total amount of data READ	TAPE READ	Library Serial Number	Transfer Size	SUM	Lifetime	Yes
TAPE_LIBRARY_WRITE_DAY	Tape Library : total amount of data WRITE	TAPE WRITE	Library Serial Number	Transfer Size	SUM	1 day	Yes

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
TAPE_LIBRARY_WRITE	Tape Library : total amount of data WRITE	TAPE WRITE	Library Serial Number	Transfer Size	SUM	Lifetime	Yes
TAPE_LAST_MOUNT_DATE	Tape: date of last MOUNT	TAPE MOUNT	Tape Barcode	Event End Time	MAX	Lifetime	Yes
TAPE_LAST_DISMOUNT	Tape: date of last DISMOUNT	TAPE DISMOUNT	Tape Barcode	Event End Time	MAX	Lifetime	Yes
TAPE_LAST_READ	Tape: date of last READ	TAPE READ	Tape Barcode	Event End Time	MAX	Lifetime	Yes
TAPE_LAST_WRITE	Tape: date of last WRITE	TAPE WRITE	Tape Barcode	Event End Time	MAX	Lifetime	Yes
TAPE_LAST_EVENT_ID	Tape: DIVAprotect Event ID of the last Tape/Drive operation	TAPE MOUNT TAPE MOUNT ERR TAPE POSITION TAPE POSITION ERR TAPE READ TAPE READ ERR TAPE WRITE TAPE WRITE ERR TAPE UNLOAD TAPE UNLOAD ERR TAPE DISMOUNT TAPE DISMOUNT	Tape Barcode	Event Id	MAX	Lifetime	Yes
TAPE_EXTERNALIZATION_NUMBER	Tape: number of externalizations	TAPE EJECT	Tape	Tape	Count	Lifetime	Yes
TAPE_REPACK_NUMBER	Tape: number of REPACK, REUSE and REFORMAT operations (together)	TAPE REPACK	Tape Barcode	Tape Barcode	Count	Lifetime	Yes
TAPE_MOUNT_NUMBER	Tape: number of MOUNT operations	TAPE MOUNT	Tape	Tape	Count	Lifetime	Yes
TAPE_READ_WRITE_NUMBER	Tape: number of READ and WRITE operations (together)	TAPE READ TAPE WRITE	Tape Barcode	Tape Barcode	Count	Lifetime	Yes
TAPE_READ_WRITE_NUMBER_DAY	Tape: number of READ and WRITE operations (together)	TAPE READ TAPE WRITE	Tape Barcode	Tape Barcode	Count	1 day	Yes
TAPE_READ_WRITE_ABORTED_NUMBER	Tape: number of aborted READ and WRITE operations (together)	TAPE READ ERR TAPE WRITE ERR	Tape Barcode	Tape Barcode	Count	Lifetime	Yes
TAPE_READ_WRITE_ABORTED_NUMBER_DAY	Tape: number of aborted READ and WRITE operations (together)	TAPE READ ERR TAPE WRITE ERR	Tape Barcode	Tape Barcode	Count	1 day	Yes
TAPE_MOUNT_DISMOUNT_ABORTED_NUMBER	Tape: number of aborted MOUNT and DISMOUNT operations (together)	TAPE MOUNT ERR TAPE DISMOUNT	Tape Barcode	Tape Barcode	Count	Lifetime	Yes
DISK_NUMBER_READ_DAY	Disk : Total number of READ operations	DISK READ DISK READ ERR	DISK NAME	Event Id	Count	1 day	Yes
DISK_NUMBER_READ	Disk : Total number of READ operations	DISK READ DISK READ ERR	DISK NAME	Event Id	Count	Lifetime	Yes
DISK_NUMBER_WRITE_DAY	Disk : Total number of WRITE operations	DISK WRITE DISK WRITE ERR	DISK NAME	Event Id	Count	1 day	Yes
DISK_NUMBER_WRITE	Disk : Total number of WRITE operations	DISK WRITE DISK WRITE ERR	DISK NAME	Event Id	Count	Lifetime	Yes
DISK_NUMBER_READ_ABORTED_DAY	Disk : Total number of ABORTED READ operations	DISK READ ERR	DISK NAME	Event Id	Count	1 day	Yes

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
DISK_NUMBER_READ_ABORTED	Disk : Total number of ABORTED READ operations	DISK READ ERR	DISK NAME	Event Id	Count	Lifetime	Yes
DISK_NUMBER_WRITE_ABORTED_DAY	Disk : Total number of ABORTED WRITE operations	DISK WRITE ERR	DISK NAME	Event Id	Count	1 day	Yes
DISK_NUMBER_WRITE_ABORTED	Disk : Total number of ABORTED WRITE operations	DISK WRITE ERR	DISK NAME	Event Id	Count	Lifetime	Yes
DISK_READ_DAY	DISK : total amount of data READ	DISK READ	DISK NAME	Transfer Size	SUM	1 day	Yes
DISK_READ	DISK : total amount of data READ	DISK READ	DISK NAME	Transfer Size	SUM	Lifetime	Yes
DISK_WRITE_DAY	DISK : total amount of data WRITE	DISK WRITE	DISK NAME	Transfer Size	SUM	1 day	Yes
DISK_WRITE	DISK : total amount of data WRITE	DISK WRITE	DISK NAME	Transfer Size	SUM	Lifetime	Yes
DISK_AVG_TRANSFER_RATE_READ_DAY	DISK : average transfer rate of READ	DISK READ	DISK NAME	Transfer Rate	Average	1 day	Yes
DISK_AVG_TRANSFER_RATE_READ	DISK : average transfer rate of READ	DISK READ	DISK NAME	Transfer Rate	Average	Lifetime	Yes
DISK_AVG_TRANSFER_RATE_WRITE_DAY	DISK : average transfer rate of WRITE	DISK WRITE	DISK NAME	Transfer Rate	Average	1 day	Yes
DISK_AVG_TRANSFER_RATE_WRITE	DISK : average transfer rate of WRITE	DISK WRITE	DISK NAME	Transfer Rate	Average	Lifetime	Yes
DISK_TIME_ALL_OPERATION_DAY	DISK : total time of ALL operations	DISK READ DISK WRITE	DISK NAME	Event Duration	SUM	1 day	Yes
DISK_TIME_ALL_OPERATION	DISK : total time of ALL operations	DISK READ DISK WRITE	DISK NAME	Event Duration	SUM	Lifetime	Yes
DISK_TIME_READ_DAY	DISK : total time of READ operations	DISK READ	DISK NAME	Event Duration	SUM	1 day	Yes
DISK_TIME_READ	DISK : total time of READ operations	DISK READ	DISK NAME	Event Duration	SUM	Lifetime	Yes
DISK_TIME_WRITE_DAY	DISK : total time of WRITE operations	DISK WRITE	DISK NAME	Event Duration	SUM	1 day	Yes
DISK_TIME_WRITE	DISK : total time of WRITE operations	DISK WRITE	DISK NAME	Event Duration	SUM	Lifetime	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_CREATED_DAY	DiwArchive System : number of objects created	ARCHIVE REQUEST TRANSCODE END COPY AS REQUEST	Local Diwa System	Event Id	Count	1 day	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_CREATED	DiwArchive System : number of objects created	ARCHIVE REQUEST TRANSCODE END COPY AS REQUEST	Local Diwa System	Event Id	Count	Lifetime	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_DELETED_DAY	DiwArchive System : number of objects deleted	DELETE OBJECT	Local Diwa System	Event Id	Count	1 day	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_DELETED	DiwArchive System : number of objects deleted	DELETE OBJECT	Local Diwa System	Event Id	Count	Lifetime	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_ARCHIVED_DAY	DiwArchive System : number of objects archived	ARCHIVE REQUEST	Local Diwa System	Event Id	Count	1 day	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_ARCHIVED	DiwArchive System : number of objects archived	ARCHIVE REQUEST	Local Diwa System	Event Id	Count	Lifetime	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_RESTORED_DAY	DiwArchive System : number of objects restored	RESTORE	Local Diwa System	Event Id	Count	1 day	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_RESTORED	DiwArchive System : number of objects restored	RESTORE	Local Diwa System	Event Id	Count	Lifetime	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_INSTANCE_CREATED_DAY	DiwArchive System : number of objects instance created	CREATE INSTANCE	Local Diwa System	Event Id	Count	1 day	Yes
DIYARCHIVE_SYSTEM_NUMBER_OBJECT_INSTANCE_CREATED	DiwArchive System : number of objects instance created	CREATE INSTANCE	Local Diwa System	Event Id	Count	Lifetime	Yes

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
DIARCHIVE_SYSTEM_NUMBER_OBJECT_INSTANCE_DELETED_DAY	DivArchive System : number of objects instance deleted	DELETE INSTANCE	Local Diva System	Event Id	Count	1 day	Yes
DIARCHIVE_SYSTEM_NUMBER_OBJECT_INSTANCE_DELETED	DivArchive System : number of objects instance deleted	DELETE INSTANCE	Local Diva System	Event Id	Count	Lifetime	Yes
DIARCHIVE_SYSTEM_NUMBER_OBJECT_INSTANCE_COPY_DAY	DivArchive System : number of objects instance copied	COPY REQUEST	Local Diva System	Event Id	Count	1 day	Yes
DIARCHIVE_SYSTEM_NUMBER_OBJECT_INSTANCE_COPY	DivArchive System : number of objects instance copied	COPY REQUEST	Local Diva System	Event Id	Count	Lifetime	Yes
DIARCHIVE_SYSTEM_READ_WRITE_NUMBER_DAY	DivArchive System : number of READ and WRITE operations	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Local Diva System	Event Id	Count	1 day	Yes
DIARCHIVE_SYSTEM_READ_WRITE_NUMBER	DivArchive System : number of READ and WRITE operations	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Local Diva System	Event Id	Count	Lifetime	Yes
DIARCHIVE_SYSTEM_READ_WRITE_DAY	DivArchive System : amount of data READ and WRITTEN	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Local Diva System	Transfer Size	SUM	1 day	Yes
DIARCHIVE_SYSTEM_READ_WRITE	DivArchive System : amount of data READ and WRITTEN	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Local Diva System	Transfer Size	SUM	Lifetime	Yes
DIARCHIVE_SYSTEM_AVG_READ_WRITE_DAY	DivArchive System : Average amount of data READ and WRITTEN	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Local Diva System	Transfer Size	Weight Average	1 day	Yes
DIARCHIVE_SYSTEM_AVG_READ_WRITE	DivArchive System : Average amount of data READ and WRITTEN	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Local Diva System	Transfer Size	Weight Average	Lifetime	Yes

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
DIARCHIVE_SYSTEM_READ_WRITE_ABORTED_NUMBER_DAY	DwArchive System : number of ABORTED READ and ABORTED WRITE operations	DISK READ ERR TAPE READ ERR SD READ ERR DISK WRITE ERR TAPE WRITE ERR SD WRITE ERR	Local Diva System	Event Id	Count	1 day	Yes
DIARCHIVE_SYSTEM_READ_WRITE_ABORTED_NUMBER	DwArchive System : number of ABORTED READ and ABORTED WRITE operations	DISK READ ERR TAPE READ ERR SD READ ERR DISK WRITE ERR TAPE WRITE ERR SD WRITE ERR	Local Diva System	Event Id	Count	Lifetime	Yes
DIARCHIVE_SYSTEM_ACTIVE_ARCHIVE_NUMBER_DAY	DwArchive System : number of active archive request	ARCHIVE REQUEST	Local Diva System	Number of Operations	MAX	1 day	Yes
DIARCHIVE_SYSTEM_ACTIVE_ARCHIVE_NUMBER	DwArchive System : number of active archive request	ARCHIVE REQUEST	Local Diva System	Number of Operations	MAX	Lifetime	Yes
DIARCHIVE_SYSTEM_ACTIVE_RESTORE_NUMBER_DAY	DwArchive System : number of active restore request	RESTORE	Local Diva System	Number of Operations	MAX	1 day	Yes
DIARCHIVE_SYSTEM_ACTIVE_RESTORE_NUMBER	DwArchive System : number of active restore request	RESTORE	Local Diva System	Number of Operations	MAX	Lifetime	Yes
DIARCHIVE_SYSTEM_ACTIVE_COPY_NUMBER_DAY	DwArchive System : number of active copy request	COPY REQUEST	Local Diva System	Number of Operations	MAX	1 day	Yes
DIARCHIVE_SYSTEM_ACTIVE_COPY_NUMBER	DwArchive System : number of active copy request	COPY REQUEST	Local Diva System	Number of Operations	MAX	Lifetime	Yes
DIARCHIVE_SYSTEM_ACTIVE_COPY_AS_NUMBER_DAY	DwArchive System : number of active copy as new object request	COPY AS REQUEST	Local Diva System	Number of Operations	MAX	1 day	Yes
DIARCHIVE_SYSTEM_ACTIVE_COPY_AS_NUMBER	DwArchive System : number of active copy as new object request	COPY AS REQUEST	Local Diva System	Number of Operations	MAX	Lifetime	Yes
DIARCHIVE_SYSTEM_OBJECT_EXPORT_NUMBER_DAY	DwArchive System : number of OBJECTS Exported	Not calculated in Phase 2	Local Diva System	Event Id	Count	1 day	Yes
DIARCHIVE_SYSTEM_OBJECT_EXPORT_NUMBER	DwArchive System : number of OBJECTS Exported	Not calculated in Phase 2	Local Diva System	Event Id	Count	Lifetime	Yes
DIARCHIVE_SYSTEM_OBJECT_INSTANCE_EXPORT_NUMBER_DAY	DwArchive System : number of INSTANCE Exported	Not calculated in Phase 2	Local Diva System	Event Id	Count	1 day	Yes
DIARCHIVE_SYSTEM_OBJECT_INSTANCE_EXPORT_NUMBER	DwArchive System : number of INSTANCE Exported	Not calculated in Phase 2	Local Diva System	Event Id	Count	Lifetime	Yes
DIARCHIVE_SYSTEM_OBJECT_IMPORT_NUMBER_DAY	DwArchive System : number of OBJECTS Imported	Not calculated in Phase 2	Local Diva System	Event Id	Count	1 day	Yes
DIARCHIVE_SYSTEM_OBJECT_IMPORT_NUMBER	DwArchive System : number of OBJECTS Imported	Not calculated in Phase 2	Local Diva System	Event Id	Count	Lifetime	Yes
DIARCHIVE_SYSTEM_OBJECT_INSTANCE_IMPORT_NUMBER_DAY	DwArchive System : number of INSTANCE Imported	Not calculated in Phase 2	Local Diva System	Event Id	Count	1 day	Yes
DIARCHIVE_SYSTEM_OBJECT_INSTANCE_IMPORT_NUMBER	DwArchive System : number of INSTANCE Imported	Not calculated in Phase 2	Local Diva System	Event Id	Count	Lifetime	Yes
MEDIA_READ_WRITE_NUMBER_DAY	Media : number of READ and WRITE operations	DISK READ TAPE READ DISK WRITE TAPE WRITE	Media	Event Id	Count	1 day	Yes

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
MEDIA_READ_WRITE_NUMBER	Media : number of READ and WRITE operations	DISK READ TAPE READ DISK WRITE TAPE WRITE	Media	Event Id	Count	Lifetime	Yes
MEDIA_READ_WRITE_DAY	Media : amount of data READ and WRITTEN	DISK READ TAPE READ DISK WRITE TAPE WRITE	Media	Transfer Size	SUM	1 day	Yes
MEDIA_READ_WRITE	Media : amount of data READ and WRITTEN	DISK READ TAPE READ DISK WRITE TAPE WRITE	Media	Transfer Size	SUM	Lifetime	Yes
MEDIA_OBJECT_INSTANCE_CREATE_DELETE_DAY	Media : number of object instance CREATED and DELETED	CREATE INSTANCE DELETE INSTANCE	Media	Event Id	Count	1 day	Yes
MEDIA_OBJECT_INSTANCE_CREATE_DELETE	Media : number of object instance CREATED and DELETED	CREATE INSTANCE DELETE INSTANCE	Media	Event Id	Count	Lifetime	Yes
MEDIA_OBJECT_INSTANCE_ONLINE_DAY	Media : number of object instance ONLINE	Not Calculated Based on Events	Media	Event Id	Count	1 day	Yes
MEDIA_OBJECT_INSTANCE_EXTERN_DAY	Media : number of object instance Externalized	Not Calculated Based on Events	Media	Event Id	Count	1 day	Yes
ACTOR_READ_WRITE_NUMBER_DAY	Actor : number of READ and WRITE operations	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Actor Name	Event Id	Count	1 day	Yes
ACTOR_READ_WRITE_NUMBER	Actor : number of READ and WRITE operations	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Actor Name	Event Id	Count	Lifetime	Yes
ACTOR_READ_WRITE_DAY	Actor : amount of data READ and WRITTEN	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Actor Name	Transfer Size	SUM	1 day	Yes
ACTOR_READ_WRITE	Actor : amount of data READ and WRITTEN	DISK READ TAPE READ SD_READ DISK WRITE TAPE WRITE SD_WRITE	Actor Name	Transfer Size	SUM	Lifetime	Yes

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
ACTOR_TIME_ALL_OPERATION_DAY	Actor : time in all operations	TAPE MOUNT TAPE MOUNT ERR TAPE POSITION TAPE POSITION ERR DISK READ DISK READ ERR TAPE READ TAPE READ ERR DISK WRITE DISK WRITE ERR TAPE WRITE TAPE WRITE ERR SD READ SD READ ERR SD WRITE SD WRITE ERR TAPE UNLOAD TAPE UNLOAD ERR END OF TAPE	Actor Name	Event Duration	SUM	1 day	Yes
ACTOR_TIME_ALL_OPERATION	Actor : time in all operations	TAPE MOUNT TAPE MOUNT ERR TAPE POSITION TAPE POSITION ERR DISK READ DISK READ ERR TAPE READ TAPE READ ERR DISK WRITE DISK WRITE ERR TAPE WRITE TAPE WRITE ERR SD READ SD READ ERR SD WRITE SD WRITE ERR TAPE UNLOAD TAPE UNLOAD ERR END OF TAPE	Actor Name	Event Duration	SUM	Lifetime	Yes
ACTOR_TIME_READ_DAY	Actor : time in READ operations	DISK READ TAPE READ SD READ	Actor Name	Event Duration	SUM	1 day	Yes
ACTOR_TIME_READ	Actor : time in READ operations	DISK READ TAPE READ SD READ	Actor Name	Event Duration	SUM	Lifetime	Yes

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
ACTOR_TIME_WRITE_DAY	Actor : time in WRITE operations	DISK WRITE TAPE WRITE SD WRITE	Actor Name	Event Duration	SUM	1 day	Yes
ACTOR_TIME_WRITE	Actor : time in WRITE operations	DISK WRITE TAPE WRITE SD WRITE	Actor Name	Event Duration	SUM	Lifetime	Yes
ACTOR_READ_WRITE_ABORTED_NUMBER_DAY	Actor : number of ABORTED READ and ABORTED WRITE operations with drives	TAPE READ ERR TAPE WRITE ERR	Actor Name	Event Id	Count	1 day	Yes
ACTOR_READ_WRITE_ABORTED_NUMBER	Actor : number of ABORTED READ and ABORTED WRITE operations with drives	TAPE READ ERR TAPE WRITE ERR	Actor Name	Event Id	Count	Lifetime	Yes
ACTOR_READ_WRITE_ABORTED_NUMBER_SD_DAY	Actor : number of ABORTED READ and ABORTED WRITE operations with SD	SD READ ERR SD WRITE ERR	Actor Name	Event Id	Count	1 day	Yes
ACTOR_READ_WRITE_ABORTED_NUMBER_SD	Actor : number of ABORTED READ and ABORTED WRITE operations with SD	SD READ ERR SD WRITE ERR	Actor Name	Event Id	Count	Lifetime	Yes
TRANSCODE_NUMBER_DAY	Transcoder : number TRANSCODE operations	TRANSCODE END	Transcoder Name	Event Id	Count	1 day	Yes
TRANSCODE_NUMBER	Transcoder : number TRANSCODE operations	TRANSCODE END	Transcoder Name	Event Id	Count	Lifetime	Yes
TRANSCODE_ABORTED_NUMBER_DAY	Transcoder : number ABORTED TRANSCODE operations	TRANSCODE ERR	Transcoder Name	Event Id	Count	1 day	Yes
TRANSCODE_ABORTED_NUMBER	Transcoder : number ABORTED TRANSCODE operations	TRANSCODE ERR	Transcoder Name	Event Id	Count	Lifetime	Yes
TRANSCODE_DATA_DAY	Transcoder : amount of data TRANSCODED	TRANSCODE END	Transcoder Name	Transfer Size	SUM	1 day	Yes
TRANSCODE_DATA	Transcoder : amount of data TRANSCODED	TRANSCODE END	Transcoder Name	Transfer Size	SUM	Lifetime	Yes
TRANSCODE_AVG_DATA_DAY	Transcoder : Average amount of data TRANSCODED	TRANSCODE END	Transcoder Name	Transfer Size	Weight Average	1 day	Yes
TRANSCODE_AVG_DATA	Transcoder : Average amount of data TRANSCODED	TRANSCODE END	Transcoder Name	Transfer Size	Weight Average	Lifetime	Yes
TRANSCODE_AVG_THROUGHPUT_DAY	Transcoder: Average transcoding throughput	TRANSCODE END	Transcoder Name	Transfer Rate	Average	1 day	Yes
TRANSCODE_AVG_THROUGHPUT	Transcoder: Average transcoding throughput	TRANSCODE END	Transcoder Name	Transfer Rate	Average	Lifetime	Yes
TRANSCODE_MIN_THROUGHPUT_DAY	Transcoder: MIN transcoding throughput	TRANSCODE END	Transcoder Name	Transfer Rate	MIN	1 day	Yes
TRANSCODE_MIN_THROUGHPUT	Transcoder: MIN transcoding throughput	TRANSCODE END	Transcoder Name	Transfer Rate	MIN	Lifetime	Yes
TRANSCODE_MAX_THROUGHPUT_DAY	Transcoder: MAX transcoding throughput	TRANSCODE END	Transcoder Name	Transfer Rate	MAX	1 day	Yes
TRANSCODE_MAX_THROUGHPUT	Transcoder: MAX transcoding throughput	TRANSCODE END	Transcoder Name	Transfer Rate	MAX	Lifetime	Yes

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
TRANSCODE_TIME_DAY	Transcoder : time in TRANSCODING operation	TRANSCODE END	Transcoder Name	Event Duration	SUM	1 day	Yes
TRANSCODE_TIME	Transcoder : time in TRANSCODING operation	TRANSCODE END	Transcoder Name	Event Duration	SUM	Lifetime	Yes
ANALYZE_NUMBER_DAY	Analyzer : number ANALYZER operations	ANALYZER END	Analyzer Name	Event Id	Count	1 day	Yes
ANALYZE_NUMBER	Analyzer : number ANALYZER operations	ANALYZER END	Analyzer Name	Event Id	Count	Lifetime	Yes
ANALYZE_ABORTED_NUMBER_DAY	Analyzer : number ABORTED ANALYZER operations	ANALYZER ERR	Analyzer Name	Event Id	Count	1 day	Yes
ANALYZE_ABORTED_NUMBER	Analyzer : number ABORTED ANALYZER operations	ANALYZER ERR	Analyzer Name	Event Id	Count	Lifetime	Yes
ANALYZE_DATA_DAY	Analyzer : amount of data ANALYZED	ANALYZER END	Analyzer Name	Transfer Size	SUM	1 day	Yes
ANALYZE_DATA	Analyzer : amount of data ANALYZED	ANALYZER END	Analyzer Name	Transfer Size	SUM	Lifetime	Yes
ANALYZE_AVG_THROUGHPUT_DAY	Analyzer : Average analyzed throughput	ANALYZER END	Analyzer Name	Transfer Rate	Average	1 day	Yes
ANALYZE_AVG_THROUGHPUT	Analyzer : Average analyzed throughput	ANALYZER END	Analyzer Name	Transfer Rate	Average	Lifetime	Yes
ANALYZE_MIN_THROUGHPUT_DAY	Analyzer : MIN analyzed throughput	ANALYZER END	Analyzer Name	Transfer Rate	MIN	1 day	Yes
ANALYZE_MIN_THROUGHPUT	Analyzer : MIN analyzed throughput	ANALYZER END	Analyzer Name	Transfer Rate	MIN	Lifetime	Yes
ANALYZE_MAX_THROUGHPUT_DAY	Analyzer : MAX analyzed throughput	ANALYZER END	Analyzer Name	Transfer Rate	MAX	1 day	Yes
ANALYZE_MAX_THROUGHPUT	Analyzer : MAX analyzed throughput	ANALYZER END	Analyzer Name	Transfer Rate	MAX	Lifetime	Yes
ANALYZE_TIME_DAY	Analyzer : time in ANALYZER operations	ANALYZER END	Analyzer Name	Event Duration	SUM	1 day	Yes
ANALYZE_TIME	Analyzer : time in ANALYZER operations	ANALYZER END	Analyzer Name	Event Duration	SUM	Lifetime	Yes
SD_READ_NUMBER_DAY	SD : number of READ operations	SD READ	SD Name	Event Id	Count	1 day	Yes
SD_READ_NUMBER	SD : number of READ operations	SD READ	SD Name	Event Id	Count	Lifetime	Yes
SD_WRITE_NUMBER_DAY	SD : number of WRITE operations	SD WRITE	SD Name	Event Id	Count	1 day	Yes
SD_WRITE_NUMBER	SD : number of WRITE operations	SD WRITE	SD Name	Event Id	Count	Lifetime	Yes
SD_READ_DAY	SD : amount of data READ	SD READ	SD Name	Transfer Size	SUM	1 day	Yes
SD_READ	SD : amount of data READ	SD READ	SD Name	Transfer Size	SUM	Lifetime	Yes
SD_WRITE_DAY	SD : amount of data WRITTEN	SD WRITE	SD Name	Transfer Size	SUM	1 day	Yes
SD_WRITE	SD : amount of data WRITTEN	SD WRITE	SD Name	Transfer Size	SUM	Lifetime	Yes
SD_TIME_DAY	SD : time in operation	SD READ SD WRITE	SD Name	Event Duration	SUM	1 day	Yes
SD_TIME	SD : time in operation	SD READ SD WRITE	SD Name	Event Duration	SUM	Lifetime	Yes
DIVAPROTECT_EXECUTION_COUNT_DAY	DIVAPROTECT : number of times DivaProtect was executed	Built In Metrics	Local Diva System	Number of Operations	Count	1 day	Yes

Table 61: Metrics Definitions

Metric Name	Metric Description	Events	Aggregate By	Collect Field	Operation	Collection Interval	Enabled
DIVAPROTECT_EXECUTION_COUNT	DIVAPROTECT : number of times DivaProtect was executed	Built In Metrics	Local Diva System	Number of Operations	Count	Lifetime	Yes
DIVAPROTECT_EVENTS_PROCESSED_DAY	DIVAPROTECT : Number of EVENTS DivaProtect Processed	Built In Metrics	Local Diva System	Number of Operations	Count	1 day	Yes
DIVAPROTECT_EVENTS_PROCESSED	DIVAPROTECT : Number of EVENTS DivaProtect Processed	Built In Metrics	Local Diva System	Number of Operations	Count	Lifetime	Yes
DIVAPROTECT_METRIC_PROCESSED_DAY	DIVAPROTECT : Number of METRICS DivaProtect Processed	Built In Metrics	Local Diva System	Number of Operations	Count	1 day	Yes
DIVAPROTECT_METRIC_PROCESSED	DIVAPROTECT : Number of METRICS DivaProtect Processed	Built In Metrics	Local Diva System	Number of Operations	Count	Lifetime	Yes
DIVAPROTECT_INTERNAL_ERROR_DAY	DIVAPROTECT : Number of DivaProtect Internal Errors	Built In Metrics	Local Diva System	Number of Operations	Count	1 day	Yes
DIVAPROTECT_INTERNAL_ERROR	DIVAPROTECT : Number of DivaProtect Internal Errors	Built In Metrics	Local Diva System	Number of Operations	Count	Lifetime	Yes

Default Configuration

Table 62: Default Configuration

Configuration Parameter	Default	Values
Manager: Enable/Disable DIVAprotect Data Collection	1	0 or 1
Manager: Size of the event batch download (<i>number of events</i>)	100	Integer
Manager: Max timeout in the event there are not events to fill the above batch (<i>seconds</i>)	15	Integer
Conf Utility GUI: Enable/Disable DIVAprotect Configuration	0	0 or 1
DB: Maximum possible number of Events in DB	1,000,000	Integer
DB: Maximum possible number of Metrics in DB	1,000,000	Integer

APPENDIX B – DIVArchive Sources and Destinations Guide

B1 Introduction

B1.1 Objective

The goal of this appendix is to provide guidelines for configuring Sources and Destinations for each type of content server supported by DIVArchive.

Refer to the *Oracle DIVarchive Supported Environments* documentation for a detailed and up-to-date list of supported content servers and formats, and related DIVArchive platforms.

B1.2 Scope

This document applies to DIVArchive 7.3. Unless modified, it should also apply to upcoming minor releases.

B2 General Items

B2.1 Introduction

This chapter introduces general items for the Source/Destination Configuration, such as:

- Features
- Configuration attributes
- Connect options

These may apply to any kind of Source/Destination, or at least to most of them (*in this case, the exact scope is specified*).

B2.2 Files Path Root

SYNOPSIS

Files Path Root is a parameter for archive/restore requests. This parameter aims at specifying the root folder for data transfers.

APPLIES TO

Any type of Source/Destination

RANGE OF VALUES

Files Path Root path can be an absolute or relative path. This parameter is limited to 260 characters.

DESCRIPTION

Each **Content Server** section of this document specifies how the Files Path Root and related File Names parameter are expected to be set for archive requests.

For (*partial*) restore requests:

- The filenames on the destination will be those specified when archiving.
- If no Files Path Root is given, then the one specified when archiving will be considered.

SEE ALSO:

Root Path: [Section B2.3](#)

B2.3 Root Path

SYNOPSIS

Root Path is a Source/Destination attribute.

This attribute may be used as a default path for ftp-like Sources/Destinations, or as a disk mount point for **Disk** and **Local** sources. This path will be appended before any Files Path Root specified in `archive/restore/...` requests, unless the path specified in a request is an absolute path (*UNIX or Windows*).

The benefit of this approach is better Source/Destination abstraction. Server directories used by DIVArchive are specified at the configuration level (*and not at the request level*), and can be changed at any time without requiring a change to DIVArchive clients.

APPLIES TO

Any type of Source/Destination

RANGE OF VALUES

Root Path will be always an absolute path (*UNIX or Windows*). An absolute path is defined as follows:

- A UNIX Absolute Path starts with '/'. For example / or /tmp.
- A Windows Absolute Path starts with letter and colon (':'); if the path length is more than 2 characters, the third character should be a backslash (\). For example; c:, c:\, C:\tmp, d:\tmp.
- Omneon Path (*player name*) is always considered as an absolute path.
- DIVArchive cannot open or create files on a Windows filesystem if their absolute path exceeds 260 characters. The Root Path must be no more than a total of 260 characters.
- Root Path may be left empty (*unspecified*) and in this instance is ignored.

DESCRIPTION

If Root Path is specified, its value is combined with the Files Path Root (*FPR*) specified in a request to give the final Source/Destination path. This is performed according to rules below:

- Relative paths are added to the absolute path, absolute paths override preceding absolute paths (*standard 'Path Arithmetic'*).

- If Root Path and FPR have different OS types (e.g. one is a UNIX path, another is Windows), the second path (FPR) is converted to the OS type specified by the first path (Root Path) (replace ‘\’ with ‘/’ and vice versa); converted path is considered as relative path.
- If Root Path ends with ‘>’ character, the FPR is always considered as a relative path; character ‘>’ is omitted during concatenation.

Table 63: Root Path Definitions

Source/Dest	Object	Request		Results		
ROOT_PATH	Original_FPR recorded in database & metadata	Request Type	FPR	Rule to be applied to create actual path to be considered for the transfer	Path to be considered for the transfer	Original_FPR recorded in database and metadata
null		Archive	null	ROOT_PATH+FPR	null	null
null			set		FPR	FPR
set			null		ROOT_PATH	null
set			set		ROOT_PATH+FPR	FPR
null		Archive with tr_arch format	null	ROOT_PATH+FPR	null	null
null			set		FPR	null
set			null		ROOT_PATH	null
set			set		ROOT_PATH+FPR	null
null	null	Restore	null	(ROOT_PATH+FPR) Original_FPR	null	
null	null		set		FPR	
set	null		null		ROOT_PATH	
set	null		set		ROOT_PATH+FPR	
null	set		null		Original FPR	
null	set		set		FPR	
set	set		null		ROOT_PATH	
set	set		set		ROOT_PATH+FPR	
	null	Transcode				null

Source/Dest	Object	Request	Results
	set	Archived	null

B2.3.1 UNIX style paths

Table 64: UNIX Style Paths

Root Path (Source/Destination)	File Path Root (request)	Actual Path to Files
/diva/upload	tmp	/diva/upload/tmp
/diva/upload	/tmp	/tmp
/diva/upload		/diva/upload
/diva/upload	C:\tmp	/diva/upload/C:/tmp (!!!)
/diva/upload>	/tmp	/diva/upload/tmp
/diva/upload>	\tmp	/diva/upload/tmp
/diva/upload>		/diva/upload

B2.3.2 Windows Style Paths

Table 65: Windows Style Paths

Root Path (Source/Destination)	File Path Root (request)	Actual path to files
D:\diva\upload	tmp	D:\diva\upload\tmp
D:\diva\upload	C:\tmp	C:\tmp
D:\diva\upload		D:\diva\upload
D:\diva\upload>	/tmp	D:\diva\upload\tmp
D:\diva\upload>	C:\tmp	D:\diva\upload\tmp
D:\diva\upload>	C:/tmp	D:\diva\upload\C:\tmp
D:\diva\upload>		D:\diva\upload

SEE ALSO:

Files Path Root.

B2.4 Metasource

SYNOPSIS

Metasource is a specific type of Source/Destination to manage several Sources/Destinations sharing the same online storage as one (*or multiple Drop Folder Monitors*) with failover and load-balancing features.

APPLIES TO

Any type of Source/Destination

DESCRIPTION

Refer to Appendix B24 Metasources for more information on the Metasource Source/Destination Type.

B2.5 Connect Options

SYNOPSIS

Connect Options are a Source/Destination parameter.

This parameter is used to specify the communication protocol with the Source/Destination or to modify the protocol's defaults.

DESCRIPTION

Some of the options exclusively apply to a specific Source/Destination Type. They are documented as part of that Source/Destination Type (*from section 3 onwards*). Some others are for wider use and are documented in this section.

Some Connect Options (*explicitly or implicitly*) specified for the Source/Destination may be superseded by those specified in `archive/restore/...` requests. This section also specifies, for each Connect Option, if it can or cannot be superseded at the request level.

B2.5.1 Quality of Service (qos=)

SYNOPSIS

`qos=`**DIRECT_AND_CACHE** | **CACHE_AND_DIRECT**

IMPORTANT: This option must be the first one in place in the Source/Destination Connect Options list. Additionally, the QOS option *must always* be specified in *lowercase*.

APPLIES TO

Any type of Source/Destination.

RANGE OF VALUES

DIRECT_AND_CACHE: Direct transfers from (or to) a Source/Destination to (or from) DIVArchive are preferred, but cache transfers will occur if it is not possible to process the request in direct mode.

CACHE_AND_DIRECT: Cache transfers from (or to) a Source/Destination to (or from) DIVArchive are preferred, but direct transfers will occur if it's not possible to process the request in cache mode.

DESCRIPTION

This option specifies the Transfer Mode used when transferring from this specific Source/Destination when archive initiator sets `QualityOfService` parameter in archive/restore parameters to `DEFAULT`.

If the Archive Initiator sets the `qualityOfService` to something other than `DEFAULT`, the QOS Connect Option is simply ignored.

Table 66: Quality of Service Connection Examples

QOS Connect Option	QualityOfService parameter set by the Archive Initiator	Actual Transfer Mode applied by DIVArchive Manager
<code>DIRECT_AND_CACHE</code>	<code>DEFAULT</code>	<code>DIRECT_AND_CACHE</code>
<code>DIRECT_AND_CACHE</code>	<code>DIRECT_ONLY</code>	<code>DIRECT_ONLY</code>
<code>DIRECT_AND_CACHE</code>	<code>CACHE_ONLY</code>	<code>CACHE_ONLY</code>
<code>CACHE_AND_DIRECT</code>	<code>DEFAULT</code>	<code>CACHE_AND_DIRECT</code>
<code>CACHE_AND_DIRECT</code>	<code>DIRECT_ONLY</code>	<code>DIRECT_ONLY</code>
<code>CACHE_AND_DIRECT</code>	<code>CACHE_ONLY</code>	<code>CACHE_ONLY</code>
	<code>DEFAULT</code>	<code>DEFAULT (i.e. DIRECT_AND_CACHE)</code>
	<code>DIRECT_ONLY</code>	<code>DIRECT_ONLY</code>
	<code>CACHE_ONLY</code>	<code>CACHE_ONLY</code>

This option cannot be superseded by the Request Option.

B2.5.2 Source/Destination FTP User Login (-login)

SYNOPSIS

Specifies the username for connection to an FTP Source/Destination.

`-login username`

APPLIES TO

When specified in Source/Destination Type description.

RANGE OF VALUES

If applicable to a Source/Destination Type, possible values are described in the related paragraph.

DESCRIPTION

This option is generally used to specify a username to connect to a Source/Destination when the transfer protocol is FTP or FTP-like.

See the paragraph dedicated to each applicable Source/Destination Type for more information.

This option can be superseded by the Request Option.

B2.5.3 Source/Destination CIFS User Login (-user)

SYNOPSIS

Specifies the username for connection to a CIFS Source/Destination.

`-user username@domain`

APPLIES TO

When specified in Source/Destination Type description.

RANGE OF VALUES

If applicable to a Source/Destination Type, possible values are described in the related paragraph.

DESCRIPTION

This option is generally used to specify a username to connect to a CIFS Source/Destination.

See the paragraph dedicated to each applicable Source/Destination Type for more information.

This option can be superseded by the Request Option.

B2.5.4 Source/Destination Password (-pass)

SYNOPSIS

Specifies the password for connection to the Source/Destination.

`-pass password`

APPLIES TO

When specified in the Source/Destination Type description.

RANGE OF VALUES

If applicable to a Source/Destination Type, possible values are described in the related paragraph.

DESCRIPTION

This option is generally used in combination with the `-login` option.

See the paragraph dedicated to each applicable Source/Destination Type for more information.

This option can be superseded by the Request Option.

B2.5.5 Source/Destination Connect Port (-port)

SYNOPSIS

Specifies the port number for connection to the Source/Destination.

`-port port`

APPLIES TO

When specified in Source/Destination Type description.

RANGE OF VALUES

Integer value.

If applicable to a Source/Destination Type, possible values are described in related paragraph.

DESCRIPTION

See paragraph dedicated to each applicable Source/Destination Type for more information.

This option can be superseded by the Request Option.

B2.5.6 Delete Source Content after Archiving (`-allow_delete_on_source`)

SYNOPSIS

Specifies if an archive request can use the **Delete on Source** QOS option.

```
-allow_delete_on_source
```

APPLIES TO

```
FTP_STANDARD
```

RANGE OF VALUES

```
-allow_delete_on_source
```

DESCRIPTION

The archive request optional parameter `delete_on_source` is used to instruct DIVArchive to delete the original asset on the source after the archive of the object has been successfully completed.

If this option is specified in an archive request and the Archive Source Type is not **LOCAL**, **DISK** or **CIFS**, the archive request will be aborted automatically by DIVArchive.

For Source Types mentioned in the *Applies to* section above it is possible to change this behavior so that archive requests will not fail when `delete_on_source` is specified in an archive request.

If the `allow_delete_on_source` option is specified, and the `delete_on_source` parameter is specified in an archive request, DIVArchive will try to delete the asset from the source once the archive has been successfully completed.

This option cannot be superseded by the Request Option.

B2.5.7 Archive/Restore File Renaming Rules (`-arch_renaming`, `-rest_renaming`)

SYNOPSIS

Rename files at archive time (`-arch_renaming`) or at restore time (`-rest_renaming`).

```
-arch_renaming <renaming_rule>+
```

```
-rest_renaming <renaming_rule>+
```

```
renaming_rule = activation_format:expression_patterns:output_format
```

APPLIES TO

These options apply to any type of Source/Destination.

RANGE OF VALUES

There are no pre-defined set of values for these options.

Option values are based on regular expressions. Possible values for these options are infinite and fully customizable.

DESCRIPTION

This feature is available for archive/restore requests.

Used when a workflow implementation requires automatic file renaming when the object is archived, when the object is (*partially*) restored, or when a transcoded object is re-archived or restored.

`-arch_renaming` option allows renaming of files during the archive process.

This option may typically be used for the following example cases:

- Need to add a file extension to archived files.
- When associated to a transcoder cache (*LOCAL Source/Destination*), archive renaming rules can be set in order to rename the files of a transcoded clip. This is useful when files created by the transcoder do not have expected names.

`-rest_renaming` option allows renaming of files during the restore process.

Typically, it will be used when Source/Destination requires strict naming of files, and the files to be transferred do not comply with these rules.

This option is available for any type of restore.

- Regular restore.
- Partial Restore (*this is an alternate way to rename Partial File Restored files*).
- N-Restore (*If multiple renaming rules are defined, DIVArchive will process the rule for each Source/Destination independently*).

B2.5.7.1 Renaming Rules Processing

At least one `renaming_rule` has to be specified for the option. All renaming rules have been moved from the Actor and Partial Restore configuration files to the Configuration Utility except the Service Name and Port.

For each file on the list to be transferred DIVArchive goes through each `renaming_rule`, starting with the first one:

- If the filename matches this rule's `activation_format`, the rule will be applied.
- If the beginning of a filename matches the evaluation condition of the first rule, the condition is satisfied.
 - For example, a condition such as `.*\ .track` will be satisfied by all of the following filenames:
 - Audio.track1
 - Audio.track2

- Video.track
- As soon as a rule is applied for a given file, other rules from the list are no longer considered.
- If none of the rules can be applied, the file is not renamed.

An `activation_format` is a regular expression (*regexp*).

Expression patterns are used to parse the filename. It is a regular expression, which will include up to 9 special symbols to identify different parts of the filename: `\1 \2 \3 \4 \5 \6 \7 \8 \9`.

`output_format` is an expression that qualifies the format of a renamed file, based on atomic items (`\1 to \9`) previously identified when applying `expression_patterns` to the original filename. Two additional specific symbols can be used:

- `\o` : stands for the object name.
- `\c` : stands for the object category.

Note: Describing how regular expression pattern matching works is beyond the scope of this document. There are many web sites on this subject such as <http://www.regular-expressions.info/>.

This option can be superseded by the Request Option.

EXAMPLES

- 1) Add `.gxf` extension to all files archived from GVG Profile (*by default, these files don't have an extension*). If a file does have an extension, the `.gxf` extension will not be added.

```
-arch_renaming <.*\..*:(.*)\.(.*):\1.\2><.*:(.*):\1.gxf>
```

This option will process filenames as follows:

Table 67: Renaming Rules Processing Examples

Input File Name	Output File Name
Star Wars	Star Wars.gxf
Readme.txt	Readme.txt
Jaws.gxf	Jaws.gxf

- 2) Remove `.gxf` extension (*if any*) at archive time

```
-arch_renaming <.*\.(.*)\.(.*):\1>
```

This option will process filenames as follows:

Table 68: Renaming Rules Processing Examples

Input File Name	Output File Name
Star Wars.gxf	Star Wars
Readme.txt	Readme.txt
Jaws.avi	Jaws.avi

3) When FlipFactory transcodes clip `foo` to Pinnacle MSS, the resulting files are named `FOO.MSS.header`, `FOO.MSS.ft`, `FOO.MSS.info`, and `FOO.MSS`. These names are not those expected by Pinnacle MSS Servers. This option fixes these discrepancies.

```
-arch_renaming  
<.*\.header:(.*)>header<.*\.ft:(.*)>ft<.*\.info:(.*)>info<.*MSS  
:(.*)>std
```

This option will process filenames as follows:

Table 69: Renaming Rules Processing Examples

Input File Name	Output File Name
<code>FOO.MSS.header</code>	header
<code>FOO.MSS.ft</code>	ft
<code>FOO.MSS.info</code>	info
<code>FOO.MSS</code>	std

SEE ALSO:

In order to help regular expression development, regular expression exercisers are available online at:

- <http://regexone.com/>
- <http://www.lornajane.net/posts/2011/simple-regular-expressions-by-example>

To use this feature, you need to know the basic regular expression syntax. Regular Expression introductory information can be found online at:

- <http://www.hathitrust.org/>
- <http://books.google.com/>
- <http://www.gutenberg.org/>

Renaming rules are associated with Source/Destination. File renaming during archive or restore is configured using the Configuration Utility.

This option cannot be superseded by the Request Option.

B2.5.8 Files to be Skipped on Restoring (-rest_ignoring)

SYNOPSIS

Ignore files at restore time.

```
-rest_ignoring <rule> <rule> <rule> <rule> ...
```

The value of this option is a set of regular expressions. Files matching one of the regular expressions will be ignored by the Source/Destination. The rule supports Unicode characters in order to offer maximum flexibility.

APPLIES TO

This option can apply to any type of Source/Destination.

RANGE OF VALUES

There are no pre-defined set of values for these options.

Option values are based on regular expressions. Possible values for these options are infinite, and fully customizable.

DESCRIPTION

This feature is available for Restore Requests, Regular Restore, Partial Restore, and N-Restore. It has been created in order to ignore some files on restore based on regular expressions. Possibilities offered by regular expressions are versatile and allow many different types of filtering.

Notes:

The files to be ignored are still read from the disk or tape instance. If the set of rules is designed to ignore all the files of an object, then no file shall be restored and the request will be completed.

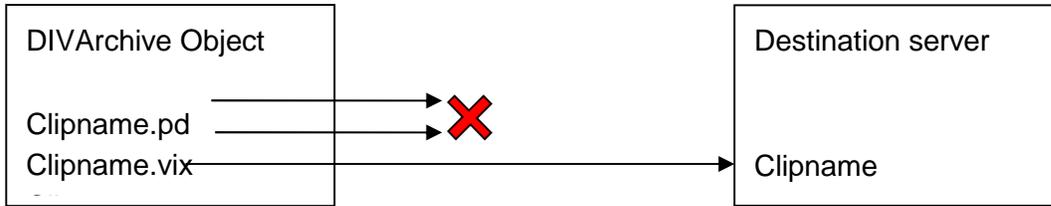
N-Restore - If multiple renaming rules are defined, DIVArchive will process the rule for each Source/Destination independently

EXAMPLE

A typical usage case is restoring a SeaChange clip to a destination that doesn't support SeaChange special files (*private data and video index files*). The option below will prevent a Source/Destination from restoring files with `.pd` or `.vix` extension:

```
-rest_ignoring <.*\.pd><.*\.vix>
```

Figure 193: Skipping Files during Restore



B2.5.9 Archive Files in a Specific Order (`-file_order`)

SYNOPSIS

Specifies files sequence during archiving or restoring.

```
-file_order MSS|OMNEON|DIFWAV|SEACHANGE DIRS_FIRST|FILES_FIRST
```

APPLIES TO

This option is not limited to specific types of Source/Destination, but is only meaningful for **LOCAL**, **DISK**, **CIFS**, and **FTP_STANDARD** Sources/Destinations.

RANGE OF VALUES

MSS: The archive sequence is:

- 1) header
- 2) ft
- 3) info
- 4) std

OMNEON: The archive sequence is:

- 1) `clip.mov`
- 2) essence files (`.wav`, `.aiff`, `.m2v`, `.mpeg`, `.diff`, etc.)

DIFWAV: The archive sequence is:

- 1) `clip.dif`
- 2) `*.wav` files

SEACHANGE: The archive sequence is:

- 1) `clip.pd`
- 2) `clip.vix` (if applicable)
- 3) `clip`

DIRS_FIRST: The archive sequence is:

```
Folder test_1;
Folder test_1\test_2;
File test_1\test_2\1.txt;
File test_1\test_2\_A2.txt;
File test_1\test_2\test.txt;
File test_1\test_2\test1.txt;
File test_1\test_2\test2.txt;
File test_1\1.txt;
File test_1\_A2.txt;
File test_1\test.txt;
File test_1\test1.txt;
File test_1\test2.txt;
File 1.txt;
File _A2.txt;
File test.txt;
File test1.txt;
File test2.txt;
```

FILES_FIRST: The archive sequence is:

```
File 1.txt;
File _A2.txt;
File test.txt;
File test1.txt;
File test2.txt;
Folder test_1;
File test_1\1.txt;
File test_1\_A2.txt;
File test_1\test.txt;
File test_1\test1.txt;
File test_1\test2.txt;
Folder test_1\test_2;
File test_1\test_2\1.txt;
File test_1\test_2\_A2.txt;
File test_1\test_2\test.txt;
File test_1\test_2\test1.txt;
File test_1\test_2\test2.txt;
```

DESCRIPTION

This option is used when files to be archived or restored are MSS files (*resp. Omneon QuickTime files*), but the source of archiving is not an AVID (*Pinnacle*) MSS Server (*resp. an Omneon server*).

This will ensure that files are archived in correct sequence, so that these files are restored in correct sequence when restoring them to a real Pinnacle MSS Server (*resp. a real Omneon server*).

DPX Partial Restore does not examine the file name or the DPX header information to determine which file is assigned to which frame. The assignment is based purely upon the order in which the `.dpx` files appear in the archive. This order, by default, is based on ordering established by the source and is typically alphanumeric. For example, NTFS DISK Source/Destinations order files and folders case insensitively as a general rule (*but not where diacritical marks, such as `ä`, `å`, `^`, etc. are applied*). When DIVArchive encounters a subfolder, by default it recursively processes all of the children of that folder (*including subfolders*) before continuing with other files. If a folder appears in the alphanumeric folder listing, it is archived recursively in the order that it appears.

However, this can create some issues – you may want all of the subdirectories of a given directory processed first, followed by the files in the directory. Or, you might want all files processed first, then subdirectories. In DIVArchive 7.0, the Actor allows the archive options `-file_order DIRS_FIRST` or `-file_order FILES_FIRST` to address these issues as described in the Range of Values shown above.

This option can be superseded by the request option.

EXAMPLE

The archive contains Seachange SAF files. These files need to be transcoded then restored to a Pinnacle MSS Server. In this case the **LOCAL** source that is used by transcoding process will be defined with option `-file_order mss` (*amongst others*).

This will ensure that files coming out of the transcoder will be archived and restored in the correct sequence, i.e. `header` first, then `ft`, `info` and finally `std`.

B2.5.10 Specifying the Transcode Format (`-tr_archive_format`, `-tr_restore_format`)

SYNOPSIS

Specifies transcode operation to apply to essence files during archive (`-tr_archive_format`) or restore (`-tr_restore_format`).

```
-tr_archive_format factory_name
```

```
-tr_restore_format factory_name
```

`factory_name`: The name of a FlipFactory factory or name of a Bitscream output format.

APPLIES TO

These options apply to any type of Source/Destination.

RANGE OF VALUES

No fixed list of values for this option.

DESCRIPTION

Each factory in a transcoder determines the format of the output file; hence, these options allow the factory (*and output format*) to be defined.

This option cannot be superseded by the Request Option except when used in a `TranscodeArchived` request.

B2.5.11 Specifying a Transcoder Name (-tr_names)

SYNOPSIS

Specifies the transcoder to be used for transcode operations.

```
-tr_names transcoder_name[transcoder_name]
```

`transcoder_name` = The name of a DIVArchive Transcoder.

APPLIES TO

These options apply to any type of Source/Destination.

RANGE OF VALUES

`transcoder_name` should be one of the transcoders defined in the Transcoders frame of the **Systems Tab** of the Configuration Utility.

DESCRIPTION

This option always comes with the option `-tr_archive_format` or the `-tr_restore_format`.

When transcoding is applied, one of the transcoders defined by the `-tr_names` option will be selected by DIVArchive according to internal Resources Allocation Policy.

If this option is not present DIVArchive will select one of the transcoders defined in the DIVArchive Configuration according to the internal Resources Allocation Policy.

This option cannot be superseded by the Request Option except when used in a `TranscodeArchived` request.

B2.5.12 Restoring Metadata (-rest_metadata or -rm)

SYNOPSIS

This option specifies that a Metadata File has to be generated and restored on every restore request.

Either form (*long or short*) of the option can be used:

`-rest_metadata`

`-rm`

APPLIES TO

This option applies to any type of Source/Destination.

Because video servers may reject the Metadata File, this option actually applies to **LOCAL**, **DISK** and **FTP_STANDARD** types.

DESCRIPTION

When the restore of an object occurs, the object is first regularly restored. After the regular restore has completed a Metadata File is generated and restored on the specified destination in the specified (*or implicit*) `FilePathRoot` of the related restore request.

Metadata File format is compliant with the DIVArchive File Set Drop Folder Metadata File specification.

Metadata Filename is: `objectname.mdf`.

B2.5.13 Restricting the Number of Actor Retries (-num_actors_to_retry)

SYNOPSIS

Specifies the maximum number of Actors on which to retry an archive, restore or Partial File Restore request.

`-num_actors_to_retry N`

`N` = number of retries.

APPLIES TO

Any type of Source/Destination.

RANGE OF VALUES

$N \geq 0$

DESCRIPTION

This option is used to limit the number of Actors that an archive, restore or Partial File Restore request will be retried upon. By default (*this option is not specified*) there is no limit and all Actors will be tried in case the request constantly fails.

This option cannot be superseded by the request option.

EXAMPLE

Option `-num_actors_to_retry 3` means that DIVArchive Manager will perform no more than 4 operations total with different Actors (*even if there are more than 4 Actors configured*).

B2.5.14 MSS Source/Destination in MXF mode (-mxf)

SYNOPSIS

Indicate when a MSS Source/Destination is configured to import/export MXF wrapped clips.

`-mxf`

The option has no parameter.

APPLIES TO

MSS Source/Destination exclusively.

DESCRIPTION

This option applies to MSS Source/Destination Type otherwise it is ignored. It is used to indicate when a MSS Source/Destination is configured to import/export MXF wrapped clips.

B2.5.15 FTP Socket Window Size (-socket_window_size)

SYNOPSIS

`-socket_window_size N`

`-socket_bufsize N` (deprecated)

`N` = buffer size in bytes.

Note: The `-socket_bufsize` syntax is still available but is not recommended after DIVArchive 6.2.2 as it may conflict with the `-socket_block_size` parameter.

APPLIES TO

This option applies to some Source/Destination Types using FTP, like `FTP_STANDARD`, `OMNEON`, `PDR`, `MSS`, etc.

RANGE OF VALUES

$2048 \geq N \geq 65536$ or more (*depends on the operating system*).

DESCRIPTION

This option is used to specify the total buffer space per-data-socket reserved for receives and sends. This parameter has a direct effect on transfer performance. Its value depends on the operating system and is usually set between 2048 and 65536 bytes. When this option is not set `DIVArchive` will use the default value set at the operating system level. It is recommended to increase this value to 32768 or more on a fast network. Some performance tests are required to identify the best setting.

Note: TCP SCALING - The TCP window Scale Option increases the TCP receive window size above its maximum value of 65,536 bytes. This option is recommended when dealing with *Long-Fat Networks*, or LFN.

B2.5.16 FTP Socket Block Size (-socket_block_size)

SYNOPSIS

`-socket_block_size N`

`N` = buffer size in kilobytes.

RANGE OF VALUES

$32 \geq N \geq 2048$

APPLIES TO

This option applies to some Source/Destination Types using FTP like `FTP_STANDARD`, `OMNEON`, `PDR`, `MSS`, etc.

DESCRIPTION

This option defines how much data (*expressed in kilobytes*) the Actor tries to send/receive with a single system call during FTP transfers. For example, if the internal buffer size of the Actor is set to 2Mb and `socket_block_size` is set to 64Kb, 32 system calls are required to write a single buffer to a data socket.

B2.5.17 FTP Passive Mode Transfers (-pasv)

SYNOPSIS

`-pasv`

`-PASV`

APPLIES TO

This option applies to some Source/Destination Types using FTP like `FTP_STANDARD`, `OMNEON`, `PDR`, `MSS`, etc.

DESCRIPTION

This option specifies that the FTP data connection should be opened in passive mode (*as opposed to active mode*) for the associated Source/Destination.

This may be necessary if a firewall is between the Actor and the Source/Destination.

B2.5.18 Restore in AXF Mode (-axf)

SYNOPSIS

Restore in AXF mode

`-axf`

APPLIES TO

This option works with the following types of Source/Destinations:

- `FTP_STANDARD`
- `SFTP`
- `LOCAL`
- `DISK`
- `EXPEDAT`

DESCRIPTION

The Restore Request optional parameter `-axf` is used to instruct DIVArchive to restore the original asset into an AXF File. Instead of purely restoring the content of an object to the destination, DIVArchive will restore the content into a new AXF File.

Combined with the `-rm` or `-rmx1` parameters, this option can be used to export an object with Metadata information and then dropped to a DFM Watch Folder.

B2.5.19 Specifying Connection Timeouts (-list_timeout, -transfer_timeout, -control_timeout)

SYNOPSIS

Specifies the maximum timeout values allowed for different connection operations and overrides the default timeout settings. The timeout value can be set for directory/file listings (*-list_timeout*), file transfers (*-transfer_timeout*), and control port connections (*-control_timeout*).

-list_timeout N

-transfer_timeout N

-control_timeout N

N = The maximum timeout allowed in seconds.

APPLIES TO

These options apply to `FTP_STANDARD`.

RANGE OF VALUES

No fixed list of values for this option.

DESCRIPTION

These options set the maximum timeout values for different FTP connection operations. If an operation exceeds the timeout value that has been set, the operation will be aborted.

The default value will be used if a timeout parameter is not used or if the timeout value is set to 0.

The default timeout values for each FTP connect operation:

-list_timeout 120 seconds

-transfer_timeout 180 seconds

-control_timeout 120 seconds

B3 Avid MSS (Program Stream)

B3.1 Introduction

Avid (formerly Pinnacle) MSS Video Servers can be installed in one of the following configurations:

- Independent Storage: the video server itself includes its own fault tolerant disk storage.
- Shared Storage: the video servers are connected to a SAN where the fault tolerant disk storage is based.

In both cases, external connectivity is provided by one, or several, Connect+ gateways supporting the FTP protocol over a Gigabit Ethernet Network.

A clip on the MSS Storage is always comprised of 3 files as listed below (or 4 if the optional information file is present):

- **header** ⇒ clip header
- **ft** ⇒ frame table
- **std** ⇒ video and audio essence
- **info** ⇒ optional information file.

All are located in a folder that matches the name of the clip (i.e. if the clip name is called FOO, the files are located in a folder also named FOO).

All files of each clip are always archived and restored by DIVArchive in the following sequence:

1. **header**
2. **ft**
3. **info**
4. **std**

Recent MediaStream Servers can export/import clips with a MXF wrapper. When configured to do MXF, this server generates a single file (*std*) which is the MXF file. In MXF Mode, there is only one file archived by DIVArchive (*std*) and it is automatically renamed to **<clipname>.mxf**. This mode is not supported by Independent Storage Servers.

B3.2 Source/Destination Configuration

B3.2.1 MSS with Independent Storage

One record is created for each MSS that DIVArchive has to move data from/to.

Table 70: MSS with Independent Storage

Attribute	Value	Example
IP Address	IP Address of the MSS	10.80.114.21
Source Type	MSS	MSS
Connection Options	System with only one gateway:	
	<code>-login gw_hostname</code>	<code>-login fcgate1</code>
	<code>-pass .video_fs</code>	<code>-pass .video_fs</code>
	System with 2 gateways:	
<code>-login gw1_hostname, gw2_hostname</code>	<code>-login fcgate1,fcgate2</code>	
<code>-pass .video_fs</code>	<code>-pass .video_fs</code>	

In a system with two gateways, `fcgate1` and `fcgate2`, DIVArchive manages failover between the two when a connect option like “`-login fcgate1, fcgate2 ...`” is specified. If the initial FTP connection fails with `fcgate1`, then it will be retried on `fcgate2`.

This feature has been deprecated and is now implemented using the `METASOURCE` Source Type.

B3.2.2 MSS with Shared Storage

One record is created for each gateway connected to the storage network that DIVArchive has to move data from/to.

Table 71: MSS with Shared Storage

Attribute	Value	Example
IP Address	IP Address of the gateway thru which the shared storage is accessed by DIVArchive.	10.80.114.28
Source Type	MSS	MSS
Connection Options	<code>-login video_fs</code> <code>-pass .video_fs</code>	<code>-login video_fs</code> <code>-pass .video_fs</code>

B3.2.3 MSS with Shared Storage (MXF mode)

One record is created for each gateway connected to the storage network DIVArchive has to move data from/to.

Table 72: MSS with Shared Storage (MXF mode)

Attribute	Value	Example
IP Address	IP Address of the gateway thru which the shared storage is accessed by DIVArchive	10.80.114.28
Source Type	MSS	MSS
Connection Options	<code>-login video_fs or mxf_fs</code> <code>-pass .video_fs or .mxf_fs</code> <code>-mxf</code>	<code>-login video_fs</code> <code>-pass .video_fs</code>

B3.3 DIVA_archiveObject Usage

Table 73: DIVA_archiveObject Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot	<code>Name_of_the_clip</code>	CITIZENKANE
FileNames	*	*

B4 Avid Airspace

B4.1 Introduction

Avid Airspace (formerly known as Pluto) is a video server with independent storage.

Each clip deals with a single essence file located on the storage root.

Airspace offers standard FTP protocol to transfer files to and from the video server internal storage over a Gigabit Ethernet Network.

B4.2 Source/Destination Configuration

One record to be created for each video server DIVArchive has to move data from/to.

Table 74: Avid Airspace Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of the video server	10.80.114.21
Source Type	FTP_STANDARD	FTP_STANDARD
Connection Options	-login ftpuser -pass FTPpasswd -port 6530	-login ftpuser -pass FTPpasswd -port 6530

B4.3 Source/Destination Usage

Table 75: Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot		
FileNames	<i>Clip name</i>	TRAFFIC

B5 Avid Transfer Manager (DHM Interface)

B5.1 Introduction

The Avid Transfer Manager is the Avid Unity Outer Gateway. It can be addressed using two different interfaces, one called the Data Handler Module (*DHM*) and the other called Dynamically Extensible Transfer (*DET*). Each interface has a specific purpose.

For this Source Type the DHM interface is used for transfer of video and audio content to/from external devices (*an archive system for instance*).

See the Avid Integration User Guide for details.

B5.2 Source/Destination Configuration

One record to be created for each video server DIVArchive has to move data from/to.

Table 76: Avid Transfer Manager (DHM interface) Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of Avid Transfer Manager	10.80.114.21
Source Type	AVID_DHM	AVID_DHM
Connection Options	<code>-port ftpPort</code> <code>-login ftpUser</code> <code>-pass ftpPass</code>	<code>-port 6021</code> <code>- login diva</code> <code>- pass diva</code>

`-port` TMCommunicator FTP service port
`-login` TMCommunicator FTP service user
`-pass` TMCommunicator FTP service user password

B5.3 Source/Destination Usage

Archive requests are initiated from Avid Edit Stations (*send to playback*). The TM Communicator supports setting custom titles for ingested (*restored*) clips. If the option `-title` is specified with a title name in a DIVArchive restore or Partial File Restore request this option value is used as the clip title, otherwise the original clip name is used. The original clip name is stored in the Video ID field of the Avid Metadata.

The following rules apply to custom title settings:

- Custom titles may consist of one or more words separated by spaces and/or tabulation characters.
- To ensure proper processing, it is strongly recommended for single word titles and absolutely required for multiword titles to be enclosed in double quotes.
- New line (`\x0A`) and carriage return (`\x0D`) characters are not allowed in titles.
- Single quote, ampersand, dash, slash, asterisk, and other special characters are supported.
- Double quote characters must be escaped with back slash to be included in the title.
- Titles composed of one or more spaces enclosed in double quotes are not considered empty.

Table 77: Avid Transfer Manager (DHM interface) Source/Destination Usage Example

Restore Option Value	Ingested Clip Title
<code>-title Clip</code>	Clip
<code>-title "Clip"</code>	Clip
<code>-title "My clip"</code>	My clip
<code>-title "My \"special\" clip"</code>	My "special" clip

B6 Avid Transfer Manager (DET Interface)

B6.1 Introduction

Avid Transfer Manager is the Avid Unity Outer Gateway. It can be addressed through two different interfaces called the Data Handler Module (*DHM*) and Dynamically Extensible Transfer (*DET*). Each interface has a specific purpose.

For this source type, the DET interface is used for transfer of Metadata and Media Files to Unity Workgroups (*or an archive system, seen as an external workgroup / Unity storage extender*).

See the Avid Integration User Guide for details.

B6.2 Source/Destination Configuration

One record to be created for each video server DIVArchive has to move data from/to.

Table 78: Avid Transfer Manager (DET Interface) Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of Avid Transfer Manager	10.80.114.21
Source Type	AVID_DET	AVID_DET
Connection Options	<code>-port ftpPort</code> <code>-login ftpUser</code> <code>-pass ftpPass</code>	<code>-port 6021</code> <code>-login det</code> <code>-pass diva</code>

`-port` TMCommunicator FTP service port
`-login` TMCommunicator FTP service user
`-pass` TMCommunicator FTP service user password

B6.3 Usage

Archive requests are initiated from Avid edit stations (*'send to workgroup'*).

B7 SeaChange BMS/BMC

B7.1 Introduction

A SeaChange BMS (*Broadcast Media Server*) is a standalone video server with its own storage. A BMS is equipped with a fast-Ethernet Interface.

A SeaChange BMC (*Broadcast Media Cluster*) is a cluster of video servers providing a unified storage based on SeaChange RAID² technology. Each server of the BMC can deliver files stored on RAID² to DIVArchive.

DIVArchive uses the SeaChange Streaming API (*SSAPI*) protocol to communicate with either a BMS or BMC using two dlls from the servers `vstrmkit` directory. It is recommended that the `seass.dll` and `RemoteVstrmAPI.dll` files be copied from the `c:\vstrmkit` directory on the SeaChange nodes, and pasted into each Actor's `bin` folder (*overwriting the old ones*). You will only be able to overwrite the files when the Actor service is stopped.

File transfer format is SAF (*SeaChange Archive Format*) only.

Note: The FTP protocol is also supported but not documented because it is now depreciated. If this functionality is required, you should contact Oracle Technical Support for more information.

B7.2 Source/Destination Configuration

By default, a SeaChange BMC Node offers Automatic Load Balancing management for data transfer across all nodes of the cluster.

If you wish to use this feature, only the last node of the BMC needs to be declared in the DIVArchive configuration. DIVArchive will then always connect to the same node of the cluster. This node will transparently redirect transfers to another node as required.

This feature can be disabled by using a special IP Address setting in the DIVArchive configuration (*see below*). In this case, **ALL** nodes of the BMC must be declared in the DIVArchive configuration.

A Metasource encompassing all nodes of the cluster may also be added to enable load balancing and failover from within DIVArchive itself.

Table 79: Seachange BMS/BMC Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of BMS or BMC Node	10.80.114.26
	It is possible to disable the SeaChange Automatic Load Balancing by placing a \$ sign in front of the IP Address of ALL BMC nodes. Syntax is \$<IP_Address>	\$10.80.114.26
Source Type	SEACHANGE_BMC	SEACHANGE_BMC
DIVAActor_SEACHANGECHECKDELAY	Identifies the delay before checking if a video was <u>not</u> deleted by SeaChange just after a restore service. The default value is 1000.	DIVAActor_SEACHANGECHECKDELAY =1000

B7.3 Source/Destination Usage

The SeaChange file system is flat. When archiving a clip, parameters have to be specified this way:

Table 80: Seachange BMS/BMC Source/Destination Usage Example

DIVA_archiveObject Parameter	Value
FilePathRoot	
FileNames	Name_of_the_clip

B8 Seachange BML

B8.1 Introduction

The SeaChange BML (*Broadcast Media Library*) is a large storage system for SeaChange Archive Format (SAF) files and is based on the RAID² technology of the SeaChange BMC platform.

DIVArchive uses the SeaChange Streaming API (SSAPI) protocol to communicate with either a BMS or BMC using two dlls from the servers `vstrmkit` folder. It is recommended that the `seass.dll` and `RemoteVstrmAPI.dll` files be copied from the `c:\vstrmkit` folder on the SeaChange nodes and pasted into each Actor's `bin` folder (*overwriting the old ones*). You will only be able to overwrite the files when the Actor service is stopped.

File transfer format is SAF (*Seachange Archive Format*) only.

Note: The FTP protocol is also supported but not documented because it is now deprecated. If this functionality is required, you should contact Oracle Technical Support for more information.

B8.2 Source/Destination Configuration

The Automatic Load Balancing feature as described for BMC also exists for BML and operates in a similar fashion.

Table 81: Seachange BML Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of the BML Node.	10.80.114.26
	It is possible to disable the SeaChange Automatic Load Balancing by placing a \$ sign in front of the IP Address of ALL BMC nodes. Syntax is \$<IP_Address>	\$10.80.114.26
Source Type	SEACHANGE_BML	SEACHANGE_BML
DIVAActor_SEACHANGECHECKDELAY	Identifies the delay before checking if a video was not deleted by SeaChange just after a restore service. The default value is 1000.	DIVAActor_SEACHANGECHECKDELAY=1000
DIRECTORY_SERVER_ENABLED	Identifies whether the BML directory server is enabled or disabled.	Valid values are 1 (<i>yes/enabled</i>) and 0 (<i>no/disabled</i>). The default value is 1 (<i>yes/enabled</i>).

B8.3 Source/Destination Usage

Seachange BML clips storage is flat. When archiving a clip, parameters have to be specified as follows:

Table 82: Seachange BML Source/Destination Usage

DIVA_archiveObject Parameter	Value
FilePathRoot	
FileNames	Name_of_the_clip

B9 Seachange BMLe/BMLex

B9.1 Introduction

The SeaChange BMLe is the storage sub-system of the latest SeaChange MediaClient architecture. More recently, the BMLe has been superseded by the BMLex series.

Both the BMLe and BMLex Servers are based on the BML architecture, however Infiniband is used for the cluster interconnect rather than the older IOP interfaces. Each node of the cluster is equipped with four FSI ports to provide high speed transfers to/from the BMLe/BMLex.

DIVArchive uses CIFS or FTP protocols to communicate with BMLe/BMLex.

File transfer format is the native format of the files stored on the BMLe/BMLex. Each asset is comprised of:

- MPEG2 Files: MPEG essence, Private Data (.PD) and Video Index (.VIX) files.
- MXF Files: MXF file (.MXF), Private Data (.PD) and Video Index (.VIX) if the MXF essence is MPEG2.

When the clip is comprised of three files (*i.e.* the essence + .VIX + .PD), these files are always archived and restored by DIVArchive in the following sequence:

1. .PD File
2. .VIX File
3. Essence File

DIVArchive can restore SAF files from the archive to the BMLe/x. When a SAF clip is restored to a BMLe/x, the SAF file is automatically unwrapped by DIVArchive and the three files are actually restored to BMLe/x (*i.e.* the essence + .PD + .VIX).

B9.2 SAF Support

This Source/Destination can restore SAF (*SeaChange Archive Format*) files from an archived SAF Object to BMLe.

This feature is transparent to the end user as DIVArchive can automatically detect SAF and unwrap it on the fly. When a SAF clip is restored to the BMLe, the SAF file is unwrapped by DIVArchive and its content is actually restored to BMLe as separate files:

- Index File (.VIX extension)
- Private Data (.PD extension)
- Essence File (*no extension*)

The name of each file is extracted from the SAF file header.

Supported SAF versions are SAF 0.1, SAF 1.0, and SAF generated by BMLe/x. The last one is particular as it may contain two consecutive Private Data files; a 12-byte PD file and a 28-byte PD file. In this case, DIVArchive will only restore the 28-byte file and the other one is ignored.

B9.3 Source/Destination Configuration

One Source/Destination to be declared for each FSI of each node:

Table 83: Seachange BMLe/BMLex Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address FSI	10.80.114.26
Source Type	SEACHANGE_BML	SEACHANGE_BML
Connection Options	-ftp -cifs -login ftpUser -user cifsUser@domain -pass password -nometadata	-cifs
DIVAActor_SEACHANGECHECKDELAY	Identifies the delay before checking if a video was not deleted by SeaChange just after a restore service. The default value is 1000.	DIVAActor_SEACHANGECHECKDELAY=1000

- ftp | -cifs One of these two options **must** be specified otherwise Streaming API protocol is assumed, which is not supported by DIVArchive for the BMLe/x.
This option cannot be superseded by the Request Option.
- ftp FTP protocol is used for data transfer to/from BMLe/x.
- cifs CIFS protocol is used for data transfer to/from the BMLe/x FSI cards. For information, the implicit CIFS path to BMLe is \\fsi_ip_address\vstrm
- nometadata This option prevents DIVArchive from archiving the .vix and .pd files when the clip to be transferred is made of essence+.vix+.pd.
This option cannot be superseded by the Request Option.

B9.4 Source/Destination Usage

When archiving a clip, parameters have to be specified this way:

Table 84: Seachange BMLe/BMLex Source/Destination Usage Example

DIVA_archiveObject Parameter	Value
FilePathRoot	
FileNames	Name_of_the_clip

B10 Leitch vR Series

B10.1 Introduction

The LEITCH VR series video server is connected to external storage that is usually shared with other video servers of the same brand.

Clips are stored on LEITCH storage as flat files, one file per clip, without any folder structure.

To move clips in and out of the shared storage, LEITCH provides a dedicated gateway called the Archive Streamer. The Archive Streamer offers standard FTP protocol over a Gigabit Ethernet Network.

B10.2 Source/Destination Configuration

Note: The LEITCH Source Type is depreciated. It was initially created to comply with the first Archive Streamer releases that did not correctly report the size of the file to be transferred.

One record to be created for each Archive Streamer DIVArchive has to move data from/to.

Table 85: Leitch vR Series Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of Leitch Archive Streamer	10.80.114.26
Source Type	FTP_STANDARD	FTP_STANDARD
Connection Options	-login ftpUser -pass ftpPass -port ftpPort	

B10.3 Source/Destination Usage

Table 86: Leitch vR Series Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot		
FileNames	Name_of_the_clip	Friends

B11 Leitch Nexio

B11.1 Introduction

The LEITCH Nexio Video Server is connected to external storage that is usually shared with other video servers of the same brand.

Clips are stored on LEITCH storage as flat files, one file per clip, without any folder structure.

Clips movement in and out of the shared storage is possible directly from the video server using the standard FTP protocol over a Gigabit Ethernet Network.

B11.2 Source/Destination Configuration

Note: The LEITCH Source Type is depreciated.

One record to be created for each video server DIVArchive has to move data from/to.

Table 87: Leitch Nexio Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of Leitch Nexio Video Server	10.80.114.26
Source Type	FTP_STANDARD	FTP_STANDARD
Connection Options	-login ftpUser -pass ftpPass -port ftpPort	

B11.3 Source/Destination Usage

Table 88: Leitch Nexio Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot		
FileNames	Name_of_the_clip	Friends

B12 Grass Valley Profile

B12.1 Introduction

Grass Valley Profile Video Servers can be provided:

- With Independent Storage: the video server includes its own fault tolerant disk storage.
- As part of a MAN: Video servers are connected to a SAN where the fault tolerant disk storage resides.

Irrespective of the storage mechanism, the DIVArchive Actor always connects to a specific Profile Server.

Exchange format is GXF only.

Profile Storage consists of one master disk (e.g. `EXT:` or `INT1:`), and one level of folders where one clip is seen as one file. One folder called `default` always exists.

Network infrastructure between GVG Profiles and DIVArchive Actors is an IP/FC network.

B12.2 Source/Destination Configuration

One record to be created for each video server DIVArchive has to move data from/to.

Table 89: Grass Valley Profile Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of the video server	10.80.114.21
Source Type	PDR	PDR
Name	Logical name for the video server	GVG-Profile-1

B12.3 Actor Configuration

The Actor configuration settings have been moved from the actor and Partial File Restore configuration files to the Configuration Utility's Actor Frame. Refer to Section 2.9.

Table 90: Grass Valley Profile Actor Configuration Parameters

Parameter	Description	Suggested Values
<code>DIVAActor_ProfileReadingBS</code>	FTP block size used for transfer on profile video servers in reading (<i>bytes</i>)	1500 16374 32768 (<i>default</i>)
<code>DIVAActor_ProfileWritingBS</code>	FTP block size used for transfer on profile video servers in writing (<i>bytes</i>)	16374 32768 (<i>default</i>)

These two parameters have a direct influence on transfer performance.

It is recommended to try several combinations of values on the target platform.

B12.4 Network Configuration

In addition to the two parameters in the previous section, the MTU size set for the HBA used for IP/FC traffic to the Profiles may have an influence on transfer performance.

Grass Valley does not provide any recommendation for MTU size.

Note: It is recommended to set the MTU size on the Actor HBA to the same value as MTU size of Profile HBA. This is only a recommended setting and not an absolute rule!

B12.5 Source/Destination Usage

When archiving a clip, parameters have to be specified as follows:

Table 91: Grass Valley Profile Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot	<i>/explodedFile/disk:/folder</i>	<i>/explodedFile/INT1:/default</i>
FileNames	<i>Name_of_the_clip</i>	MyClip

B13 Grass Valley UIM

B13.1 Introduction

UIM is a gateway to Profile Servers (*standalone or MAN*). They provide TCP/IP over Gigabit Ethernet connections to external systems (*such as DIVArchive*).

For legacy purposes, the connection can also be IP/FC for regular Profiles.

UIM also provides on-the-fly format conversion (*to MXF at least*).

Exchange format with UIM is GXF by default, or alternately MXF.

B13.2 Source/Destination Configuration

One record to be created for each UIM DIVArchive has to move data from/to.

Table 92: Grass Valley UIM Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of the UIM	10.80.114.21
Source Type	PDR	PDR
Connection Options	<code>-login movie mxfmovie ...</code> <code>-format ?D10AES3 ...</code> <code>-extension ...</code>	<code>-login mxfmovie</code> <code>-format ?D10AES3</code> <code>-extension .mxf</code>

-login

Specifies FTP user for login onto the UIM in order to achieve transfers in desired format.

Known logins are:

movie	GXF exchange format
mxfmovie	MXF exchange format

If the `-login` option is not specified, the user **movie** is assumed.

-format

UIM supported options for some file formats (*depends on -login option*).

The only known option is:

<code>?D10AES3</code>	e-VTR compliant file format To be used in combination with « <code>-login mxfmovie</code> » option. If this option is not specified (<i>default</i>), MXF files will be GrassValley <code>OP1a</code> format
-----------------------	--

This option can be superseded by the Request Option.

-extension

This option adds the specified extension to the original clip name in the archive.

Example: If the original clip is `clip1` and the `-extension .mxf` option is specified, then the archived file will be `clip1.mxf`.

Suppress the specified extension before restoring to the destination if it exists.

Example: If the archived file is `clip1.mxf` and `-extension .mxf` option is specified, the restored file on the destination will be `clip1`.

This option is deprecated and has been replaced by the `-arch_renaming` and the `-rest_renaming` options.

This option can be superseded by the Request Option.

B13.3 Actor Usage

UIM are gateways to the Profile. Usage is the same for UIM and Profiles. Refer to Appendix B12.5 (*whether the file is transferred in GXF or MXF makes no difference*).

B14 Grass Valley K2

B14.1 Introduction

- From DIVArchive's standpoint, K2 servers are like Profiles and UIM's combined.
- K2 servers offer Gigabit Ethernet connections to external systems.
- Exchange format with K2 is GXF by default and alternately MXF.

B14.2 Source/Destination Configuration

One record to be created for each K2 server that DIVArchive has to move data from/to.

Table 93: Grass Valley K2 Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of the K2 Server	10.80.114.21
Source Type	PDR	PDR
Main Connection Options	<code>-k2</code> <code>-login movie mxfmovie ...</code> <code>-format ?D10AES3 ...</code> <code>-extension ...</code>	<code>-k2</code> <code>-login mxfmovie</code> <code>-format ?D10AES3</code> <code>-extension .mxf</code>

-k2

Specifies the interface with K2 Servers.

If this option is set, DIVArchive will retrieve the size of the file to be transferred before the actual archive transfer (*K2 FTP does support SIZE command*). Correct transfer progress will then be reported by DIVArchive.

If this option is not set DIVArchive will assume that servers are Profile, and will not retrieve the file size before archive transfers. Progress will then stay at 0% before suddenly jumping to 100% when the transfer is complete.

This option has no impact on transferred content.

This option can be superseded by the Request Option.

-login

Specifies FTP User for login onto the K2 Server in order to achieve transfers in the desired format.

Known logins so far are:

Movie	GXF exchange format
mxfmovie	MXF exchange format

If the `-login` option is not specified, the user **movie** is assumed.

-format

K2 supported options for some file formats (*depends on -login option*).

The only know option is:

<code>?D10AES3</code>	e-VTR compliant file format To be used in combination with «-login mxfmovie» option. If this option is not specified (<i>default</i>), MXF files will be GrassValley oP1a format
-----------------------	--

This option can be superseded by the request option.

-extension

Add specified extension to the original clip name in the archive.

Example: If the original clip is `clip1` and `-extension .mxf` option is specified, then the archived file will be named `clip1.mxf`.

Suppress specified extension before restoring to the destination if exists.

Example: If the archived file is `clip1.mxf` and `-extension .mxf` option is specified, the restored file on the destination will be named `clip1`.

This option is deprecated and has been replaced by the `-arch_renaming` and `-rest_renaming` options.

This option can be superseded by the Request Option.

B14.3 Actor Usage

Usage is the same for K2 and Profiles. Refer to Appendix B12.5 (*whether the file is transferred in GXF or MXF makes no difference*).

B15 Grass Valley M-Series iVDR

B15.1 Introduction

Grass Valley iVTR is an analogue/digital VTR that includes a Gigabit connection for material exchange as GXF files.

iVTR exchange protocol is similar to the exchange protocol for Profiles.

B15.2 Source/Destination Configuration

One record to be created for each video server DIVArchive has to move data from/to.

Table 94: Grass Valley M-Series iVDR Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of the iVTR	10.80.114.21
Source Type	PDR	PDR
Name	Logical name for the iVTR	GVG-iVTR

B15.3 Source/Destination Usage

When archiving a clip, parameters have to be specified as follows:

Table 95: Grass Valley M-Series iVDR Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot	<i>/explodedFile/disk:/folder</i>	<i>/explodedFile/INT1:/default</i>
FileNames	<i>Name_of_the_clip</i>	MyClip

B16 Sony MAV70

B16.1 Introduction

The Sony MAV70 video server has its own independent storage. MAV70 storage organization is flat and all files reside in the storage root.

A Linux machine in front of each MAV70 provides a standard FTP connection for moving data to and from the video server over a Gigabit Ethernet Network.

B16.2 Source/Destination Configuration

One record to be created for each MAV70 gateway that DIVArchive has to move data from/to.

Table 96: Sony MAV70 Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of the MAV70 gateway	10.80.114.21
Source Type	FTP_STANDARD	FTP_STANDARD
Main Connection Options	-login wing -pass mpegworld	-login wing -pass mpegworld

B16.3 Source/Destination Usage

When archiving a clip, parameters have to be specified this way:

Table 97: Sony MAV70 Source/Destination Usage Example

DIVA_archiveObject Parameter	Value
FilePathRoot	
FileNames	Name_of_the_clip

B17 Omneon Spectrum MediaDirector (Quicktime)

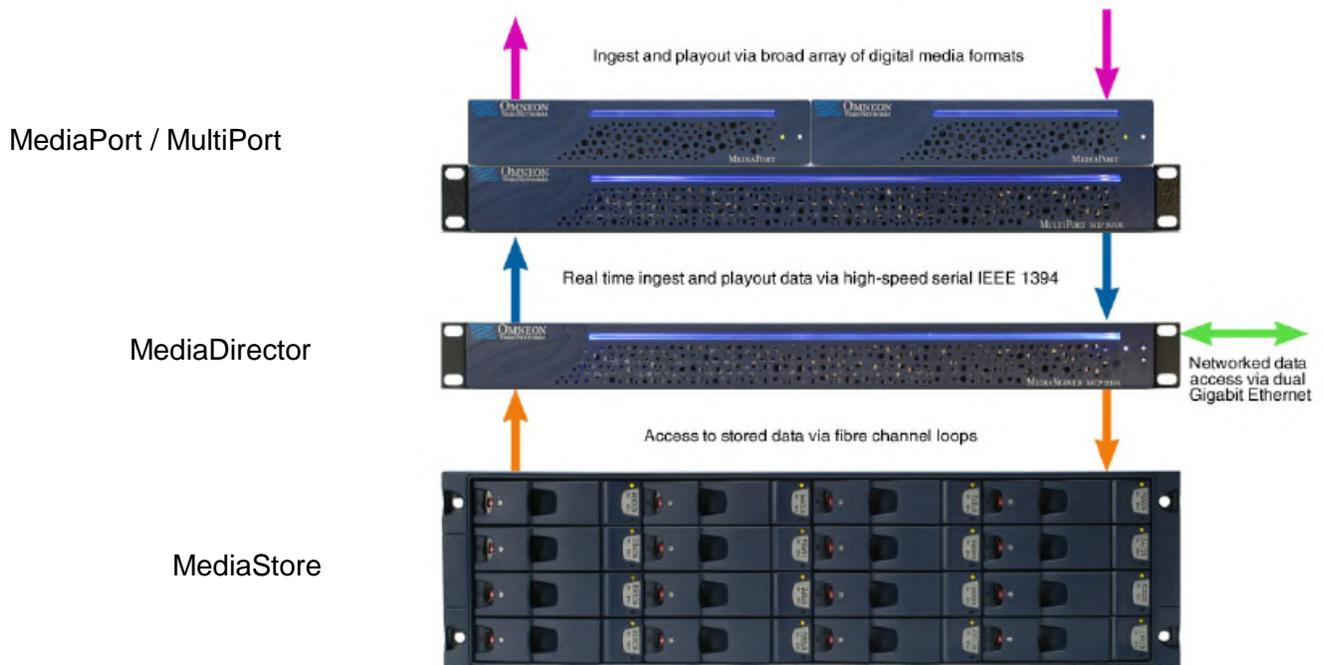
B17.1 Introduction

The Omneon MediaDirector is the heart of the Omneon Spectrum architecture. It is connected to MediaPorts or MultiPorts which handle isochronous ingest/payout, and to external storage that is usually shared with other Omneon MediaDirectors.

External storage may be MediaStore or the new MediaGrid architecture.

This section addresses MediaDirector connecting to MediaStore (for MediaGrid support in DIVArchive).

Figure 194: Omneon Spectrum MediaDirector Connections



DIVArchive interfaces with an Omneon MediaDirector to move clips in and out of the shared storage, using standard FTP protocol, over a Gigabit Ethernet Network.

When Omneon Spectrum Servers are configured to ingest material in QuickTime format, essence files are stored in a specific folder structure. Unique FTP site commands are used by the DIVArchive Actors for smart and transparent access to essence files (*in particular, the automatic discovery of a folders structure and collision-avoidance at restore time*).

B17.2 Source/Destination Configuration

One record is created for each MediaDirector that DIVArchive has to move data from/to.

Table 98: Omneon Spectrum MediaDirector Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of Omneon Director	10.80.114.21
Source Type	OMNEON	OMNEON
Root Path	To be left empty or absolute clip directory.	/default/clip.dir
Main Connection Options	-streaming_mode -sm -tempdir_mode	-streaming_mode -sm -tempdir_mode

-streaming_mode (or -sm)

This option is QuickTime specific and has no effect on the MXF content. If this option is set, DIVArchive will restore the QuickTime reference file in the following order:

- Audio tracks first
- QuickTime file
- Video track

The restore workflow is specific when this option is set. The temporary folder is used by DIVArchive to cache the QuickTime file.

-tempdir_mode

This option is used to perform partial file restores of MXF files and is applicable only to Omneon servers. The MXF partial file restore request will abort if this option is not included in the request.

B17.4 Source/Destination Usage

Table 99: Omneon Spectrum MediaDirector Source/Destination Archive Parameters

DIVA_archiveObject Parameter	Value	Example
FilePathRoot	Absolute clip folder or can be empty to use the configured Root Path	/default/clip.dir
FileNames	clip_name	myclip

Table 100: Omneon Spectrum MediaDirector Source/Destination Restore Parameters

DIVA_restoreObject Parameter	Value	Example
FilePathRoot	Absolute clip folder or can be empty to use the configured Root Path	<code>/default/clip.dir</code>

B18 Omneon Mediagrid

B18.1 Introduction

The MediaGrid is the new Content Storage System from Omneon to which Omneon Spectrum Servers can be connected.

The two major components of a MediaGrid System are:

- ContentServer(s): these store and provide access to media.
- ContentDirector(s): these act as overall file system controllers and manage the distribution of data throughout the system.

Like any other client system, DIVArchive gets access to the media via a MediaGrid ContentDirector. DIVArchive interfaces with MediaGrid using the CIFS protocol exclusively over a Gigabit Ethernet Network.

The MediaGrid ContentDirector manages data access while the data transfer occurs directly to/from the ContentServers. The Omneon File System Driver (*FSD*), installed on MediaGrid clients hides this complexity to client systems.

The Omneon FSD has to be installed on each Actor that has to exchange material with MediaGrid.

The latest version of Omneon FSD for Windows 2003 is available for download at:

<http://support.omneon.com/Updates/Omneon/Current/MediaGrid/WinFSD>

The password (*if required*) is `alloyparka`.

When material is wrapped in QuickTime format, the essence files are stored using a specific folder structure. DIVArchive also uses unique FTP site commands for smart and transparent access to the essence files (*in particular, automatic discovery of folders structure and collision-avoidance at restore time*).

Note: When the Actor is running as a Windows Service, MediaGrid shares are accessed through a UNC path because drive letters mapped to network drives are not accessible by Windows Services.

In this case, ensure that:

- Omneon MediaGrid folders to be accessed by DIVArchive are properly shared for a given Windows User.
- DIVArchive Actor Service is configured to run under this user account.
- The user has local administrative rights on the DIVArchive Actor.

B18.2 Source/Destination Configuration

One record is required for each ContentDirector DIVArchive has to move data from/to.

Table 101: ContentDirector Attributes

Attribute	Value	Example
IP	To be left empty	

Attribute	Value	Example
Address		
Source Type	MEDIAGRID	MEDIAGRID
Root Path	\\ContentDirector\filesystem\clip.dir	\\10.30.0.200\cldev4\clip.dir \\mycontentdir\fs5\clip.dir

B18.3 Source/Destination Usage

Table 102: Omneon Mediagrid Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot		
FileNames	clip_name	myclip

In cases where the material is wrapped as QuickTime, DIVArchive will search for a file matching `clip_name.mov` or `clip_name.MOV`. DIVArchive will automatically retrieve and process transfers all of the potentially referenced files.

In instances where the material is wrapped as MXF, DIVArchive will search for a file matching `clip_name.mxf` or `clip_name.MXF`. There is only one file per clip.

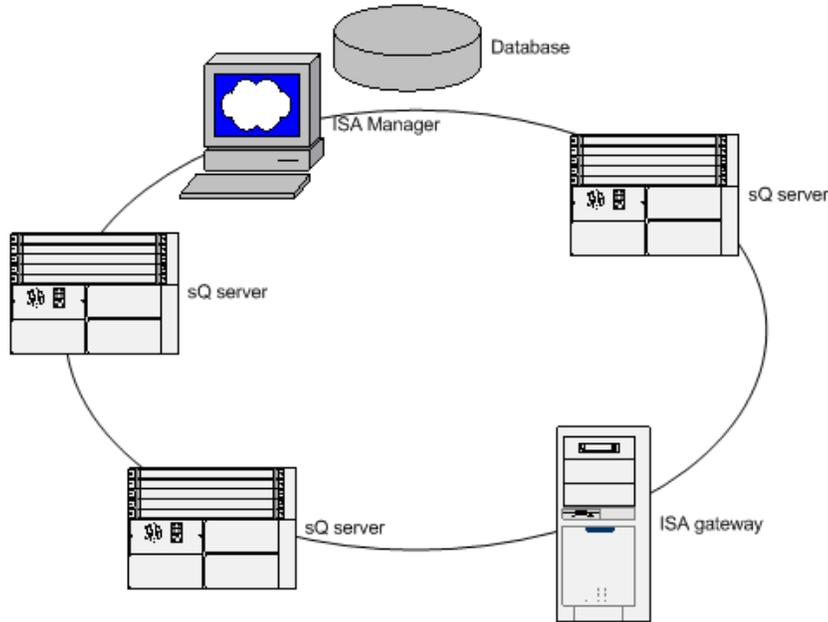
B19 Quantel Power Portal

B19.1 Introduction

The Quantel Power Portal was formerly known as the ISA Gateway.

Shown below is a view of ISA Architecture Components:

Figure 195: Quantel Power Portal Overview



An ISA System consists of:

- An ISA Manager that contains the Clip Database. Clips are identified with a unique FID (*File Identifier*) in the ISA System.
- One or more Q or sQ Servers. These servers contain video cards and disk arrays. Each disk array is associated to a `POOL ID`, and a single sQ Server can have several `POOL IDs`.

Table 103: Quantel Power Portal Sample System

sQ Server ID	POOL IDs
1	1,2
2	3
3	4

- An ISA gateway (*Power Portal*). This gateway is a FTP Server that is used to import and export clips.

B19.2 Source/Destination Configuration

One record is to be created for each Quantel ISA Gateway that DIVArchive has to move data from/to.

Table 104: Quantel Power Portal Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of the video server	10.80.114.21
Source Type	QUANTEL_ISA	QUANTEL_ISA
Connection Options	-login ftpUser -pass ftpPass	

B19.3 Actor Configuration

The Actor configuration settings have been moved from the actor and Partial File Restore configuration files to the Configuration Utility.

Table 105: Quantel Power Portal Actor Configuration Parameters

Parameter	Description	Suggested values
DIVAActor_QUANTELRENAMECLIPS	Enables/disables File Renaming feature (see below)	0 = renaming feature is disabled 1 = renaming feature is enabled

DIVAActor_QUANTELRENAMECLIPS applies to restore requests only.

If this parameter is set to 1, and the Object Name format is `clipName, UID (Omnibus naming)`, then object related files will be renamed using `clipName` as the Name Root.

Example: If object `superman,01AB45` is composed of files `8152.D10` and `8152.WAV` and is restored to a `QUANTEL_ISA` Destination, then:

- If `DIVAActor_QUANTELRENAMECLIPS` is set to 0, DIVArchive will transfer files called `8152.D10` and `8152.WAV` to Power Portal.
- If `DIVAActor_QUANTELRENAMECLIPS` is set to 1, DIVArchive will transfer files called `superman.D10` and `superman.WAV` respectively to Power Portal.

B19.4 Source/Destination Usage

ARCHIVE

Quantel storage is flat. When archiving a clip the parameters have to be specified as follows:

Table 106: Quantel Power Portal Source/Destination Archive Operation Usage

DIVA_archiveObject Parameter	Value
FilePathRoot	
FileNames	FID1.extension1[,FID1.extension2,...]

Files coming from Power Portal can be of different types, including:

- **D10+WAV** : filenames will be similar to 8152.D10 and 8152.WAV
- **MXF**: TestClip.mxf
- **TAR**: FramesDifference.tar

RESTORE

If a file is restored twice to Power Portal, the first file is not overwritten. The second restore will create a new file that is identified by a new FID. The DIVArchive Actor will capture this new FID at the end of the transfer and forward it to the DIVArchive Manager.

In order to get the new FID using the DIVARCHIVE API, `DIVA_GetRequestInfo` must be called. If the request is completed, the new FID will be in the additional request info field within `clipID` tags. The `clipID` tag is also encapsulated in a higher level tag called `ADDITIONAL_INFO`.

```
<ADDITIONAL_INFO>
  <ClipID>8546</ClipID>
</ADDITIONAL_INFO>
```

Automation is also free to specify a `POOL ID` in the `FilePathRoot` restore request parameter. In case no `POOL ID` is specified, Power Portal will automatically assign one at restore time.

B20 Sony Hyper Agent

B20.1 Introduction

Hyper Agent is the name given to Newsbase's FTP Server from SONY. The implementation of this FTP server is specific because `LIST` returns a proprietary formatted list of files. This list contains duration and start/end time codes but not the size of the file in bytes. The size of each clip is calculated by the Actor using three values; duration, frame rate and bitrate. The resultant size is not accurate but it is enough for the Manager to allocate a tape for all archive requests. The progress bar is not affected by this approximated size.

Duration, frame rate and bitrate are retrieved using the following two commands:

`SITE FSIZ <clip ID>` - This `SITE` command returns the duration of the specified clip.

`SITE GCNF` - Returns the current system configuration of the server.

Note: This system configuration must remain the same to make sure that all of the clips on the server are the same.

These two commands are set by the Actor at the beginning of each archive request.

Below is an example log entry of communication between Actor and the Hyper Agent FTP:

```
SITE FSIZ 1444247
200 150                <- Duration is 150 frames
SITE GCNF
213-System configureation
PAL                    <- Framerate is 25fr/s
20
30.0                   <- Bitrate is 30Mb/s
D10
SD_IFRAMEONLY
213 End of system configuraion
```

Note: In the log displayed above, the word 'configuration' is misspelled; this is a bug of the FTP server.

B20.2 Source/Destination Configuration

One record is created for each ClipBox DIVArchive has to move data from/to.

Table 107: Sony Hyper Agent Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of Newsbase Server	10.80.114.21
Source Type	SONY_HYPER_AGENT	SONY_HYPER_AGENT
Connection Options	-login <i>username</i> -pass <i>password</i>	-login sony -pass sony

B20.3 Source/Destination Usage

Table 108: Sony Hyper Agent Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilesPathRoot		
FileNames	Clip ID	1444247

B21 Standard FTP/SFTP Servers

B21.1 Introduction

DIVArchive can interface with any regular FTP Server. This can be UNIX or NT-based FTP Servers. Video servers supporting a fully RFC-959 compliant FTP Server are also considered **Standard FTP Servers**. Only one restriction applies: UNIX directory listing style is required.

When dealing with Microsoft IIS FTP Server, this parameter can be set in the **Home Directory** section of the FTP Site Properties.

DIVArchive can also interface with **SFTP (SSH FTP)** Servers.

B21.2 Source/Destination Configuration

One record is created for each video server DIVArchive has to move data from/to.

Table 109: Standard FTP/SFTP Servers Source/Destination Attributes

Attribute	Value	Example
IP Address	IP Address of the video server	10.80.114.21
Source Type	FTP_STANDARD	FTP_STANDARD
	SFTP	
Connection Options	<code>-login username</code> <code>-pass password</code> <code>-port portNumber</code>	<code>-login moon</code> <code>-pass mars</code> <code>-port 27</code>

-login

FTP/SFTP user.
Default value: anonymous

-pass

FTP/SFTP user password
Default value: anonymous

-port

FTP/SFTP port.
Default value for **FTP_STANDARD**: 21
Default value for **SFTP**: 22

B21.3 Source/Destination Usage

When archiving a clip, parameters can be specified in three different ways:

Table 110: Standard FTP/SFTP Servers Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot	Full Path to the files	/my_videos/movies
	Partial Path to the files	/my_videos
	No Path at all	
FileNames	Names of the files	maniolia, matrix
	Partial Path and Names of the files	movies/maniolia, movies/matrix
	Full Path and Names to the files	/my_videos/movies/maniolia, /my_videos/movies/matrix

DISK_FTP_PASSIVE_MODE

By default, data connections are created in active mode. This means that the DivaFtp client connects from a random unprivileged port ($N > 1023$). Then, it starts listening to this port and sends a `PORT` command to the FTP server. Valid values for this parameter are 0 (*disabled*) and 1 (*enabled*).

By setting `DISK_FTP_PASSIVE_MODE` to 1 (*enabled*), data connections are created in passive mode which means DivaFtp will send a `PASV` command and the socket is created by the server.

DISK_FTP_BLOCK_SIZE

`DISK_FTP_BLOCK_SIZE` defines how much data Actor tries to send/receive with a single system call during FTP transfers. For example, if the internal buffer size of Actor is set to 2Mbytes and `DISK_FTP_BLOCK_SIZE` is set to 32768 bytes, 64 system calls are required to write a single buffer to a data socket. The default value is 32768 bytes.

DISK_FTP_SOCKET_WINDOW_SIZE

`DISK_FTP_SOCKET_WINDOW_SIZE` is an option to adjust the normal buffer sizes allocated for output and input buffers. `DISK_FTP_SOCKET_WINDOW_SIZE` is internally used to set `so_SNDBUF` and `so_RCVBUF` for FTP managed disk types. The default value is 65536.

B22 Local

B22.1 Introduction

A local source represents a disk partition for a specific Actor (*internal disks, NAS or SAN disks*).

A local source is tied to specific Actor (*vs. Disk Source that is not tied to any particular Actor*).

B22.2 Source/Destination Configuration

One record is created for each local source that DIVArchive has to move data from/to.

Table 111: Local Source/Destination Attributes

Attribute	Value	Example
Name	Same name as the Actor this source is bound to.	<code>actor1</code>
IP Address	Same IP Address as the Actor this source is bound to.	<code>10.80.114.21</code>
Source Type	LOCAL	LOCAL

B22.3 Source/Destination Usage

When archiving a clip, parameters can be specified in three different ways:

Table 112: Local Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot	Full Path to the files	<code>/my_videos/movies</code>
	Partial Path to the files	<code>/my_videos</code>
	No Path at all	
FileNames	Names of the files	<code>maniolia, matrix</code>
	Partial Path and Names of the files	<code>movies/maniolia</code> <code>movies/matrix</code>
	Full Path and Names of the files	<code>/my_videos/movies/maniolia</code> <code>/my_videos/movies/matrix</code>

Note: If NT drive letters (e.g. "E:") are used, it is highly recommended to leave them in the `FilesPathRoot` section (i.e. use *scheme 1 or 2 above*). Including them in the `FileNames` section will prevent the request from replacing them with another path at restore time, thereby preventing the restoring of these objects on a different platform, like a Unix-like FTP Server, where drive letters are not considered valid paths.

B23 Disk / CIFS

B23.1 Introduction

A DISK or CIFS source represents a disk partition assumed to be visible from all Actors of the Production System.

The only difference between DISK and CIFS is the way blocks of data are read/written:

- DISK will instruct Actors to use (*Windows*) direct I/O.
- CIFS will instruct Actors to use (*Windows*) buffered I/O.
- DISK/CIFS sources support UNC paths.

B23.2 Source/Destination Configuration

One record is created for each Disk Source that DIVArchive has to move data from the Source to the Destination.

A generic source may also be created to represent any kind of DISK/CIFS source.

Table 113: Disk/CIFS Source/Destination Attributes

Attribute	Value	Example
Name	Nickname for Disk Source	generic-disk
IP Address		
Source Type	DISK	
	CIFS	

B23.3 Source/Destination Usage

When archiving a clip, parameters can be specified in three different ways:

Table 114: Disk / CIFS Source/Destination Usage Example

DIVA_archiveObject Parameter	Value	Example
FilePathRoot	Full Path to the files	<code>/my_videos/movies</code>
	Partial Path to the files	<code>/my_videos</code>
	No Path at all	
FileNames	Names of the files	<code>maniolia, matrix</code>
	Partial Path and Names of the files	<code>movies/maniolia</code> <code>movies/matrix</code>
	Full Path and Names to the files	<code>/my_videos/movies/maniolia</code> <code>/my_videos/movies/matrix</code>

Note: If NT drive letters (e.g. "E:") are used, it is highly recommended to leave them in the `FilePathRoot` section (i.e. use scheme 1 or 2 above). Including them in the `FileNames` section will prevent the request from replacing them with another path at restore time thereby preventing the restoring of these objects on a different platform, like a Unix-like FTP Server, where drive letters are not considered valid paths.

B24 Metasources

B24.1 Introduction

A Metasource is a collection of several (*single*) Sources of the same type. It is assumed that all Sources of the Metasource are sharing the same online storage.

Each Source of the Metasource should be of the same regular type (*i.e. any type except METASOURCE*), aka Metasource Base Type.

A Metasource provides load-balancing and failover mechanisms across all single sources of the Metasource.

B24.2 Source/Destination Configuration

One record to be created for each Metasource DIVArchive has to move data from/to.

Table 115: Metasource Source/Destination Attributes

Attribute	Value	Example	Comment
Name	Name for video servers shared storage.	gvg-man-production	
IP Address	server1[,server2,...]	pdr1,pdr2,pdr3	server1, server2, ... also have to be defined in the configuration as regular sources of the same type (<i>all types except METASOURCE, LOCAL, and DISK are permitted, e.g. OMNEON, PDR, etc.</i>)
Source Type	METASOURCE		
Production System	Has to be the same for Metasource and all Single Sources.		Manager will not start if no match.
Site	Either one or the other of the sites from Metasource Single Sources.		Site specified for Metasource will be considered by Manager for resource selection.
Root Path	A root path may be specified at Metasource level.		If Metasource Root Path is null, Root Path from selected Single Source will be considered.

Attribute	Value	Example	Comment
Max Accesses	Actual value for Metasource doesn't matter.		Value from selected Single Source will be considered. These fields cannot be left empty; suggestion is to set traffic regulation parameters to the sum of all Single Sources respective parameters. It is recommended that no changes to this parameter be made while there are active requests; it can lead to request abortion.
Max Write Acc.			
Max Read Acc.			
Max Throughput			
Main Connection Options	-failover_time=f -retry_actor=r	-failover_time=300 -retry_actor=3	

-failover_time=f

f = number of milliseconds

When a Single Source has been selected to process a request and failed, the Single Source will temporarily not be considered as part of the Metasource during 600 milliseconds.

This default value can be changed with this option.

This option cannot be superseded by the request option.

-retry_actor=r

r = integer

Used to specify the number of Metasource Single Sources to be tried for each Actor that can be part of the request processing. Default for **r** (*when option is not specified*) is 2.

Example:

If the Metasource is defined as {sd1, sd2, sd3}, and if the set of possible Actors is {a1, a2} and if **r** is set to 2, then DIVArchive will try 4 combinations maximum: most likely a0-sd1, a0-sd2, a1-sd3, a1-sd1.

This option cannot be superseded by the Request Option.

Other Single Source connection options may be also specified for the Metasource. The table below shows what the effect is for each possible option when it is specified at the Metasource level:

Table 116: Metasource Source/Destination Connection Options

Connection Option	Considered?	Comment
<code>qos=</code>	NO	<code>qos</code> value should be the same for all Metasource Single Sources, otherwise Manager will not start.
<code>-login</code>	NO	Value from selected Single Source will be considered. Applicable to FTP Source/Destinations.
<code>-user</code>	NO	Value from selected Single Source will be considered. Applicable to CIFS Source/Destinations.
<code>-pass</code>	NO	Value from selected Single Source will be considered.
<code>-port</code>	NO	Value from selected Single Source will be considered.
<code>-allow_delete_on_source</code>	NO	Implicitly assumed to be true if all Single Sources (<i>implicitly or explicitly</i>) allow deleting on Source. Otherwise assumed to be false.
<code>-arch_renaming</code>	NO	Value from selected Single Source will be considered.
<code>-rest_renaming</code>	NO	Value from selected Single Source will be considered.
<code>-file_order</code>	NO	Value from selected Single Source will be considered.
<code>-tr_archive_format</code>	YES	Values specified for Single Sources don't matter.
<code>-tr_restore_format</code>	YES	Values specified for Single Sources don't matter.
<code>-tr_names</code>	YES	Values specified for Single Sources don't matter.
<code>-rest_metadata</code>	NO	Value from selected Single Source will be considered.
<code>-num_actors_retry=</code>	YES	Values specified for Single Sources don't matter.
<code>-ftp</code>	NO	Value from selected Single Source will be considered.
<code>-cifs</code>	NO	Value from selected Single Source will be considered.
<code>-nometadata</code>	NO	Value from selected Single Source will be considered.

Connection Option	Considered?	Comment
<code>-format</code>	NO	Value from selected Single Source will be considered.
<code>-extension</code>	NO	Value from selected Single Source will be considered.
<code>-k2</code>	NO	Value from selected Single Source will be considered.

B24.3 Source/Destination Usage

A Metasource is used exactly the same way as any source of Metasource Base Type.

B24.4 Archive Request with Delete on Source (-r and -delete_fpr)

There are instances where it is required to delete content and possibly the parent folder on a server. In order to satisfy all possible scenarios there are two options available:

- `-r` (*recursive*)
- `-delete_fpr` (*includes deletion of a parent folder*)

The two options, `-r` and `-delete_fpr`, work either separately, or together, as described in the following workflow examples:

Example 1:

FilePathRoot: `C:\source\root`

Files: `*`

Options: `-r`

Result: DIVArchive will:

- Delete the content of `C:\source\root` recursively.

Example 2:

FilePathRoot: `C:\source\root`

Files: `*`

Options: `-r -delete_fpr`

Result: DIVArchive will:

- Delete the content of `C:\source\root` recursively
- Delete `root`.

Example 3:

FilePathRoot: C:\source\root

Files: *

Options:

Result: DIVArchive will:

- Delete the content of C:\source\root (*first level only*).

Example 4:

FilePathRoot: C:\source\root

Files: *

Options: -delete_fpr

Result: DIVArchive will:

- Delete the content of C:\source\root (*first level only*)
- Eventually delete root if it is empty.

Example 5:

FilePathRoot: C:\source\root

Files: object*

Options: -r

Result: DIVArchive will:

- Delete the content of C:\source\root\object recursively.
- Delete C:\source\root\object.

Example 6:

FilePathRoot: C:\source\root

Files: object*

Options: -r -delete_fpr

Result: DIVArchive will

- Delete the content of C:\source\root\object recursively.
- Delete C:\source\root\object.
- Delete C:\source\root if empty.

Example 7:

FilePathRoot: C:\source\root

Files: object1*

object2*

Options: -r

Result: DIVArchive will

- Delete the content of `C:\source\root\object1` recursively.
- Delete `C:\source\root\object1`.
- Delete the content of `C:\source\root\object2` recursively.
- Delete `C:\source\root\object2`.

Example 8:

FilePathRoot: `C:\source\root`

Files: `object1*`
`object2*`

Options: `-r -delete_fpr`

Result: DIVArchive will

- Delete the content of `C:\source\root\object1` recursively.
- Delete `C:\source\root\object1`.
- Delete the content of `C:\source\root\object2` recursively.
- Delete `C:\source\root\object2`.
- Delete `C:\source\root` if it is empty.

Example 9:

FilePathRoot: `C:\source\root`

Files: `object1*`
`object2\subfolder\clip.mov`

Options: `-r -delete_fpr`

Result: DIVArchive will

- Delete the content of `C:\source\root\object1` recursively.
- Delete `C:\source\root\object1`.
- Delete the content of `C:\source\root\object2\subfolder\clip.mov`.
- Delete `C:\source\root\object2\subfolder` if it is empty.
- Delete `C:\source\root\object2` if it is empty.
- Delete `C:\source\root` if it is empty.

B25 Expedat

B25.1 Introduction

DIVArchive can interface with DataExpedition Expedat Servers; also known as servedat. This solution utilizes MTP, which is a high performance file transfer protocol. This WAN acceleration software can utilize 100% of the bandwidth of any long distance or high latency networks.

Refer to the DataExpedition Expedat Server Installation Manual for detailed information on installation and configuration.

B25.2 Source/Destination Configuration

One record is created for each Expedat Server that DIVArchive must move data from/to.

Table 117: Source Destination Configuration

Attribute	Value	Example
IP Address	IP Address of the Expedat Server.	10.80.114.21
Source Type	EXPEDAT	EXPEDAT
Connection Options	<code>-login username</code> <code>-pass password</code> <code>-port portNumber</code> <code>-license licenseCode</code> <code>-encryption</code> <code>-seq_buffer_size <Mbytes></code> <code>-exp_maxrate <Kbytes></code> <code>-exp_mindatagram <Bytes></code>	<code>-login moon</code> <code>-pass ph4!hi4</code> <code>-port 8080</code> <code>-license 46FE464A98</code> <code>-encryption</code> <code>-seq_buffer_size 16</code> <code>-exp_maxrate 1024</code> <code>-exp_mindatagram 2848</code>

Table Notes:

- `-login` and `-pass` are mandatory if the server is configured with authentication parameters.
- `-port` should be present since there is no default value.
- `-license` is mandatory.
- `-encryption` is optional.
- `-seq_buffer_size <Mbytes>`
 - Defines the size of the DataExpedition internal buffer for each transfer. The default value is 16MB and should be sufficient for most transfers. A large buffer allows DataExpedition to continue moving data during times when the sender or receiver may not be able to process it. A small buffer will consume less memory.

- `-exp_maxrate <Kbytes>`
 - This option sets an approximate limit on the number of kilobytes per second, per transfer. The default is unlimited but can be used as an alternate method of controlling bandwidth.
- `-exp_mindatagram <Bytes>`
 - This transfer protocol is over UDP. This option can define a minimum size for each network datagram payload that DataExpedition will send. The purpose is to prevent DataExpedition from sending too small of a packet over the network. Users may wish to set this value between 2848 and 8544 when using a very fast network path (*Gigabit or higher*) and every device along the path supports Jumbo Frames (*MTU 9000*). Using large datagrams may greatly reduce CPU overhead; however use of this setting **without** Jumbo Frames being fully supported may cause severe performance issues or loss of connectivity.

B25.2.1 Enable Expedat Transfer Encryption (-encryption)

SYNOPSIS

Enables Expedat Encryption during transfers.

`-encryption`

APPLIES TO

This option works with the EXPEDAT Source/Destination.

DESCRIPTION

This option enables Expedat's content encryption during transfers.

B25.2.2 Expedat License Code (-license)

SYNOPSIS

Expedat license code.

`-license`

APPLIES TO

This option works with the EXPEDAT Source/Destination.

DESCRIPTION

This is a mandatory parameter to be able to use the DIVArchive Expedat Client. Without the license code the EXPEDAT Source/Destination is unusable.

Note: Only 1 Expedat license key can be configured per production system.

B25.3 Source /Destination Usage

This Source/Destination works similar to the FTP_STANDARD S/D in terms of the `Files Path Root` and list of files.

When Expedat Server is configured with folders having the `RestrictHome` setting enabled, the Root Path for the Data Expedition Source/Destination entry should not reference an absolute path. The Root Path may be interpreted as a path that is not accessible from the Expedat Home Directory. For example, the Root Path `/` is interpreted as `c:\`, but if the Expedat Home Directory is `D:\folder`, Expedat will attempt to access the path `D:\folder` on `c:\` - which is not valid. However, if the home directory is `c:\folder`, using the Root Path `/` is acceptable.

Instead of using an absolute path, relative path addressing must be used to resolve this situation. Relative Path Addressing is accomplished by:

1. Making the Root Path entry empty in the Configuration Utility.
2. Specifying the relative path in the File Root Path of the GUI Manager or API request for the Archive/Restore Operation.

To setup a default home location such that an API request can always use "" files path, the Expedat `cv_password.txt` file should contain a login account assigned to a folder with the `RestrictHome` option set.

Example:

```
diva:diva:::S:\DFM:RestrictHome
diva1:diva:::S:\DFM1:RestrictHome
diva2:diva:::S:\some_other_folder:RestrictHome
```

The separate user login/password accounts will allow for the creation of more than one EXPEDAT Source/Destination entries with different home locations. The API request can then reference the EXPEDAT Source/Destination pointing to the desired home location.

APPENDIX C – ADIC SDLC Installation Guide

C1 SDLC Server

C1.1 Prerequisites

The SDLC Server Process is called `supervisor`. The SDLC GUI is also available as an applet in your web browser address bar by typing in the IP Address of the computer on which SDLC Server is running.

You should avoid stopping the SDLC Manager (*i.e. the Windows Service NobleNet PortMapper for TCP – see next section*) while SDLC Clients are currently connected (*e.g. the SDLC GUI connection*). In the event this occurs, the SDLC Server will vary to a transient state making it temporarily impossible to restart.

C1.2 Configuration

To make the SDLC usable, a physical resources partition must first be defined (*in the SDLC GUI **Libraries** tab, then the **Wizard** tab*). Once this is done, a logical library is defined with its slots and drives. When the wizard completes, the partition is automatically bound to an ADIC Client. The client is used by the DIVArchive ADIC Robot Manager to get status information about library items and to send mount/dismount commands.

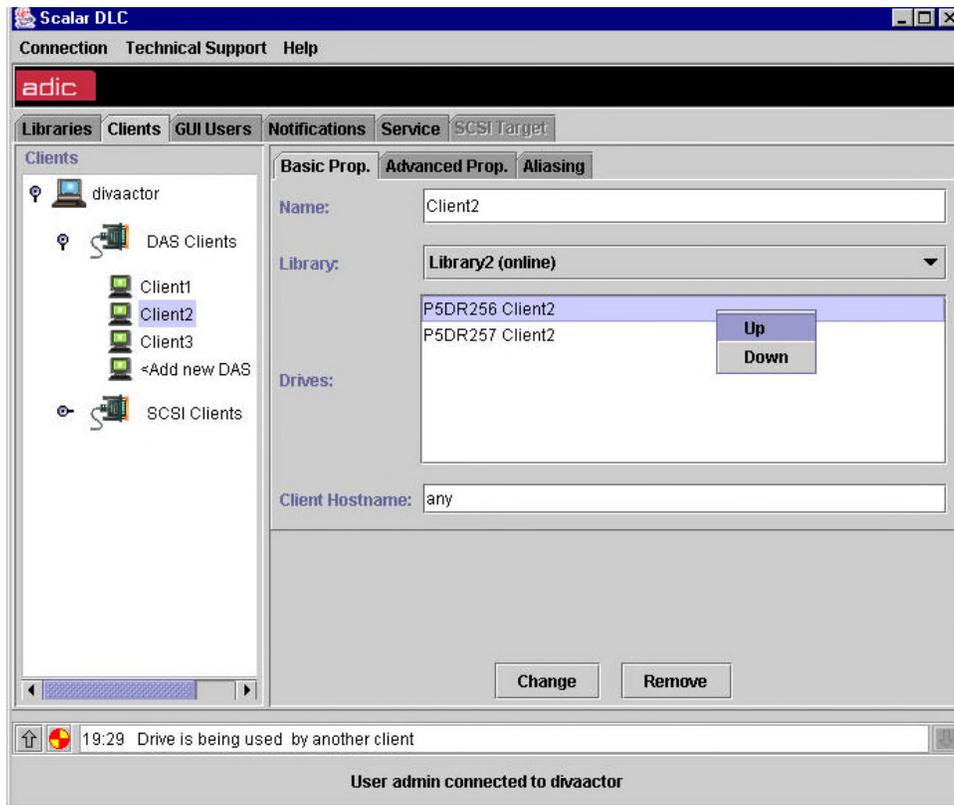
Check that the drives that will be used by DIVArchive are bound to the client dedicated to the DIVArchive ADIC Robot Manager.

In the figure below, two drives are bound to the client: `client2`. To bind a drive to a specific client:

- Select the client
- Right-click on the drive
- Select **Up** from the menu

Once a drive is bound to a client, the name of the client is appended to the name of the drive.

Figure 196: SDLC GUI - Binding a Drive to a Client.



Check that for the given client (*client2* in the above example), the field called `client hostname` is configured with the Hostname or IP Address of the client computer (*i.e. the client that will use the client2 connection when communicating with the SDLC Server*).

It is possible to specify **any** in this field in order to accept any incoming connections from any client computer that will provide `client2` as the client name when connecting through the SDLC API. Use **any** in order to use the `dasadmin` tool from a supervising computer.

Check that the Windows Service `NobleNet PortMapper for TCP` is also started. If it is not, start the service.

C2 SDLC Client

C2.1 Prerequisites

The SDLC client must be installed on the computer where the DIVArchive ADIC Robot Manager is installed.

C2.2 Installation

Install the SDLC Client from the SDLC distribution. During installation, you will be prompted for the name of the client that will be used by the ADIC Robot Manager to connect to SDLC Server. In this case, you should use the client that you created in the SDLC Server. In our example, this would be `client2`.

Note: The Client Name is case sensitive.

Check that the Windows Service `NobleNet PortMapper for TCP` is started. If it is not, start the service.

C2.3 Configuration

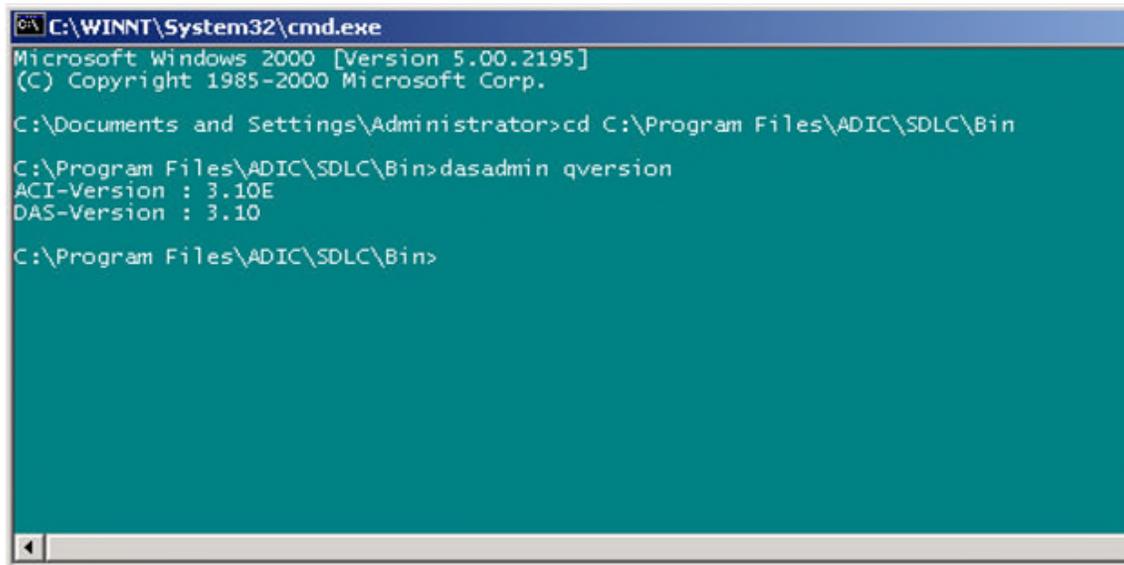
Two Windows Environmental Variables have to be defined on a Windows system as follows (*right-click on **My Computer**, select **Properties**, then the **Advanced** tab, then **Environment Variables***).

Table 118: SDLC Environment Variables

Environment variable	Definition	Example
<code>DAS_SERVER</code>	Hostname or IP Address of the computer on which the SDLC Server has been installed.	10.201.10.100
<code>DAS_CLIENT</code>	Name of the Client that the DIVArchive RobotManager will use to connect to SDLC	<code>Client2</code>

Test the SDLC Client connection to the SDLC Server in a command window using `dasadmin` (see *example and troubleshooting* section below).

Figure 197: Testing the SDLC Client/Server Connection



```
C:\WINNT\System32\cmd.exe
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\Documents and Settings\Administrator>cd C:\Program Files\ADIC\SDLC\Bin

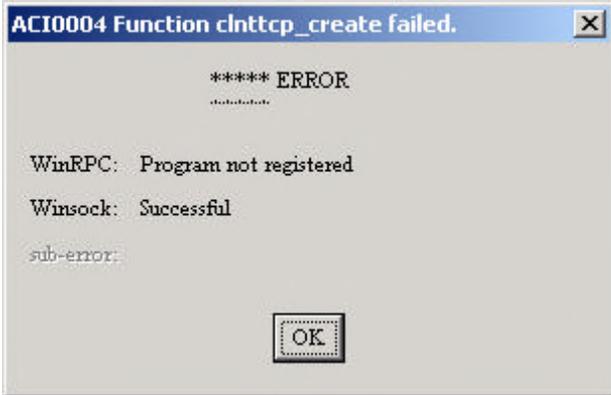
C:\Program Files\ADIC\SDLC\Bin>dasadmin qversion
ACI-Version : 3.10E
DAS-Version : 3.10

C:\Program Files\ADIC\SDLC\Bin>
```

C3 Troubleshooting

`dasadmin qversion` may not answer as shown in the figure above. The most common cases and remedies are described below.

Table 119: Common Errors and Resolutions

Error	Troubleshooting
<p>The following window appears:</p>  <p>And in the command window:</p> <pre>version failed: An RPC failure occurred. ACI-Version : 3.10E DAS error = 1</pre>	<p>On the server, check that it is possible to connect to that client from the machine on which <code>dasadmin</code> was launched.</p>
<pre>version failed: Invalid hostname or IP Address ACI-Version : 3.10E DAS error = 14</pre>	<p>On the server, check that it is possible to connect to that client from the machine on which <code>dasadmin</code> was launched. The <code>client hostname</code> is probably set to <code>localhost</code>.</p>
<pre>version failed: Invalid pointer to IDAS interface ACI-Version : 3.10E DAS error = 28</pre>	<p>Check the <code>DAS_CLIENT</code> Environment Variable setting.</p>
<p>Command never ends</p>	<p>Check that SDLC Server is started.</p> <p>Check <code>DAS_SERVER</code> Environment Variable setting.</p> <p>Check that the NT Service <code>NobleNet PortMapper for TCP</code> is started.</p>

C3.1 Useful *dasadmin* Commands

C3.1.1 Getting *dasadmin* help

```
C:\Program Files\ADIC\SDLC\Bin>dasadmin -h
```

C3.1.2 Mounting a tape

```
dasadmin mount 000160 [drive]
```

Where 000160 is the Tape ID. If `drive` is not specified then the first free drive will be automatically chosen.

C3.1.3 Dismounting a tape

```
dasadmin dism 000160
```

Where 000160 is the ID of the tape to be dismounted.

OR

```
dasadmin dism P3DR256
```

Where P3DR256 is the name of the drive to be dismounted.

Note: Tape must first be ejected with a SCSI unload before dismounting.

C3.1.4 Ejecting a Tape

```
dasadmin eject2 000160 E01
```

Where 000160 is the name of the tape to be ejected and E01 is the name of the eject/insert area.

Note: Depending upon the server configuration, the eject/insert area (*i.e. slots from the CAP*) can have different names.

C3.1.5 Inserting a Tape

```
dasadmin insert2 -n E01
```

Where `-n` specifies Data Tapes (*for Cleaning Tapes, use `-c`*) and E01 is the name of the eject/insert area.

Note: Depending on the server configuration, the eject/insert area (*i.e. slots from the CAP*) can have different names.

C3.1.6 Querying Drives

```
dasadmin ld
```

C3.1.7 Retrieving the Tapes List

```
dasadmin qvolsrange "" "" 1000
```

(Gives a list of 1000 tapes in the library).

C3.1.8 Parking the Robot Arm

```
dasadmin robhome R1
```

C3.1.9 Synchronizing the SDLC Database and Library

```
dasadmin inventory
```

C3.1.10 Retrieving Tape Information

```
dasadmin view 000160
```

C3.1.11 Dasadmin Version Info

```
dasadmin qversion
```

C3.1.12 Library Configuration Information

```
dasadmin eif_conf
```

Note: Not supported in SDLC 2.1+

C3.1.13 Dasadmin References

See the file `sd1c_doc.pdf` on the SDLC Installation CDROM.