User Guide

Version 9.4.0

Revision 1

telestream

Copyrights and Trademark Notices

Copyright © 2024 Telestream, LLC and its Affiliates. All rights reserved. No part of this publication may be reproduced, transmitted, transcribed, altered, or translated into any languages without written permission of Telestream, LLC. Information and specifications in this document are subject to change without notice and do not represent a commitment on the part of Telestream. Specifications subject to change without notice.

Telestream, Aurora, CaptionMaker, CaptureVU, Cerify, Content Manager, ContentCentral, Cricket, DIVA, DIVAdirector, DIVADocs, DIVAGrid, DIVANet, DIVAProtect, DIVASolutions, Episode, Encoding Intelligence, Episode, FLEXVU, Flip4Mac, FlipFactory, Flip Player, Geminus, Glim, GraphicsFactory, Inspector, IQ & Design, Kumulate, Lightspeed, MassStore, MassTech, MetaFlip, Post Producer, Prism, ScreenFlow, Sentry, Singulus, Split-and-Stitch, Stay Genlock, Surveyor, Tempo, TrafficManager, Vantage, Vantage Cloud Port, VOD Producer, and Wirecast are registered trademarks of Telestream, LLC and its Affiliates and its Affiliates.

Argus, ContentAgent, Cricket, e-Captioning, Inspector, iQ, iVMS, iVMS ASM, MacCaption, Pipeline, Switch, and Vidchecker are trademarks of Telestream, LLC and its Affiliates. All other trademarks are the property of their respective owners.

Adobe. Adobe® HTTP Dynamic Streaming Copyright © 2014 Adobe Systems. All rights reserved.

Apple. QuickTime, MacOS X, and Safari are trademarks of Apple, Inc. Bonjour, the Bonjour logo, and the Bonjour symbol are trademarks of Apple, Inc.

Avid. Portions of this product Copyright 2012 Avid Technology, Inc.

CoreOS. Developers of ETCD.

Dolby. Dolby and the double-D symbol are registered trademarks of Dolby Laboratories Licensing Corporation.

Fraunhofer IIS and Thomson Multimedia. MPEG Layer-3 audio coding technology licensed from Fraunhofer IIS and Thomson Multimedia.

intoPIX and Fraunhofer. Notice under 35 U.S.C. §287(a): This product or service includes JPEG XS compliant features that are covered by patents in the United States and in other jurisdictions owned by intoPIX SA (“intoPIX”) and/or Fraunhofer-Gesellschaft zur Foerderung der angewandten Forschung E.V. (“Fraunhofer”) and listed at HYPERLINK "http://www.jpegxspool.com"www.jpegxspool.com. Additional patents may be pending in United States and elsewhere.

Google. VP6 and VP8 Copyright Google Inc. 2014 All rights reserved.

MainConcept. MainConcept is a registered trademark of MainConcept LLC and MainConcept AG. Copyright 2004 MainConcept Multimedia Technologies.

Manzanita. Manzanita is a registered trademark of Manzanita Systems, Inc.

MCW. HEVC Decoding software licensed from MCW.

MediaInfo. Copyright © 2002-2013 MediaArea.net SARL. All rights reserved.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Microsoft. Microsoft, Windows NT|2000|XP|XP Professional|Server 2003|Server 2008 |Server 2012|Server 2016|Server 2019|Server 2022, Windows 7, Windows 8, Windows 10, Windows 11, Media Player, Media Encoder, .Net, Internet Explorer, SQL Server 2005|2008|2012|2016|2019, and Windows Media Technologies are trademarks of Microsoft Corporation.

NLOG, MIT, Apache, Google. NLog open source code used in this product under MIT License and Apache License is copyright © 2014-2016 by Google, Inc., © 2016 by Stabzs, © 2015 by Hiro, Sjoerd Tieleman, © 2016 by Denis Pushkarev, © 2015 by Dash Industry Forum. All rights reserved.

SharpSSH2. SharpSSH2 Copyright (c) 2008, Ryan Faircloth. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer:

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Diversified Sales and Service, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Swagger. Licensed from SmartBear.

Telerik. RadControls for ASP.NET AJAX copyright Telerik All rights reserved.

VoiceAge. This product is manufactured by Telestream under license from VoiceAge Corporation.

x264 LLC. The product is manufactured by Telestream under license from x264 LLC.

Xceed. The Software is Copyright ©1994-2012 Xceed Software Inc., all rights reserved.

ZLIB. Copyright (C) 1995-2013 Jean-loup Gailly and Mark Adler.

Other brands, product names, and company names are trademarks of their respective holders, and are used for identification purpose only.

MPEG Disclaimers

MPEGLA MPEG2 Patent

ANY USE OF THIS PRODUCT IN ANY MANNER OTHER THAN PERSONAL USE THAT COMPLIES WITH THE MPEG-2 STANDARD FOR ENCODING VIDEO INFORMATION FOR PACKAGED MEDIA IS EXPRESSLY PROHIBITED WITHOUT A LICENSE UNDER APPLICABLE PATENTS IN THE MPEG-2 PATENT PORTFOLIO, WHICH LICENSE IS AVAILABLE FROM MPEG LA, LLC, 4600 S. Ulster Street, Suite 400, Denver, Colorado 80237 U.S.A.

MPEGLA MPEG4 VISUAL

THIS PRODUCT IS LICENSED UNDER THE MPEG-4 VISUAL PATENT PORTFOLIO LICENSE FOR THE PERSONAL AND NON-COMMERCIAL USE OF A CONSUMER FOR (i) ENCODING VIDEO IN COMPLIANCE WITH THE MPEG-4 VISUAL STANDARD ("MPEG-4 VIDEO") AND/OR (ii) DECODING MPEG-4 VIDEO THAT WAS ENCODED BY A CONSUMER ENGAGED IN A PERSONAL AND NON-COMMERCIAL ACTIVITY AND/OR WAS OBTAINED FROM A VIDEO PROVIDER LICENSE IS GRANTED OR SHALL BE IMPLIED FOR ANY OTHER USE. ADDITIONAL INFORMATION INCLUDING THAT RELATING TO PROMOTIONAL, INTERNAL AND COMMERCIAL USES AND LICENSING MAY BE OBTAINED FROM MPEG LA, LLC. SEE HTTP://WWW.MPEGLA.COM.

MPEGLA AVC

THIS PRODUCT IS LICENSED UNDER THE AVC PATENT PORTFOLIO LICENSE FOR THE PERSONAL USE OF A CONSUMER OR OTHER USES IN WHICH IT DOES NOT RECEIVE REMUNERATION TO (i) ENCODE VIDEO IN COMPLIANCE WITH THE AVC STANDARD ("AVC VIDEO") AND/OR (ii) DECODE AVC VIDEO THAT WAS ENCODED BY A CONSUMER ENGAGED IN A PERSONAL ACTIVITY AND/OR WAS OBTAINED FROM A VIDEO PROVIDER LICENSED TO PROVIDE AVC VIDEO. NO LICENSE IS GRANTED OR SHALL BE IMPLIED FOR ANY OTHER USE. ADDITIONAL INFORMATION MAY BE OBTAINED FROM MPEG LA, L.L.C. SEE HTTP://WWW.MPEGLA.COM.

MPEG4 SYSTEMS

THIS PRODUCT IS LICENSED UNDER THE MPEG-4 SYSTEMS PATENT PORTFOLIO LICENSE FOR ENCODING IN COMPLIANCE WITH THE MPEG-4 SYSTEMS STANDARD, EXCEPT THAT AN ADDITIONAL LICENSE AND PAYMENT OF ROYALTIES ARE NECESSARY FOR ENCODING IN CONNECTION WITH (i) DATA STORED OR REPLICATED IN PHYSICAL MEDIA WHICH IS PAID FOR ON A TITLE BY TITLE BASIS AND/OR (ii) DATA WHICH IS PAID FOR ON A TITLE BY TITLE BASIS AND IS TRANSMITTED TO AN END USER FOR PERMANENT STORAGE AND/OR USE. SUCH ADDITIONAL LICENSE MAY BE OBTAINED FROM MPEG LA, LLC. SEE HTTP://WWW.MPEGLA.COM FOR ADDITIONAL DETAILS.

Limited Warranty and Disclaimers

Telestream, LLC (the Company) warrants to the original registered end user that the product will perform as stated below for a period of one (1) year from the date of shipment from factory:

Hardware and Media—The Product hardware components, if any, including equipment supplied but not manufactured by the Company but NOT including any third party equipment that has been substituted by the Distributor for such equipment (the "Hardware"), will be free from defects in materials and workmanship under normal operating conditions and use.

Warranty Remedies

Your sole remedies under this limited warranty are as follows:

Hardware and Media—The Company will either repair or replace (at its option) any defective Hardware component or part, or Software Media, with new or like new Hardware components or Software Media. Components may not be necessarily the same, but will be of equivalent operation and quality.

Software Updates

Except as may be provided in a separate agreement between Telestream and You, if any, Telestream is under no obligation to maintain or support the Software and Telestream has no obligation to furnish you with any further assistance, technical

support, documentation, software, update, upgrades, or information of any nature or kind.

Restrictions and Conditions of Limited Warranty

This Limited Warranty will be void and of no force and effect if (i) Product Hardware or Software Media, or any part thereof, is damaged due to abuse, misuse, alteration, neglect, or shipping, or as a result of service or modification by a party other than the Company, or (ii) Software is modified without the written consent of the Company.

Limitations of Warranties

THE EXPRESS WARRANTIES SET FORTH IN THIS AGREEMENT ARE IN LIEU OF ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. No oral or written information or advice given by the Company, its distributors, dealers or agents, shall increase the scope of this Limited Warranty or create any new warranties.

Geographical Limitation of Warranty—This limited warranty is valid only within the country in which the Product is purchased/licensed.

Limitations on Remedies—YOUR EXCLUSIVE REMEDIES, AND THE ENTIRE LIABILITY OF TELESTREAM, LLC WITH RESPECT TO THE PRODUCT, SHALL BE AS STATED IN THIS LIMITED WARRANTY. Your sole and exclusive remedy for any and all breaches of any Limited Warranty by the Company shall be the recovery of reasonable damages which, in the aggregate, shall not exceed the total amount of the combined license fee and purchase price paid by you for the Product.

Damages

TELESTREAM, LLC SHALL NOT BE LIABLE TO YOU FOR ANY DAMAGES, INCLUDING ANY LOST PROFITS, LOST SAVINGS, OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF YOUR USE OR INABILITY TO USE THE PRODUCT, OR THE BREACH OF ANY EXPRESS OR IMPLIED WARRANTY, EVEN IF THE COMPANY HAS BEEN ADVISED OF THE POSSIBILITY OF THOSE DAMAGES, OR ANY REMEDY PROVIDED FAILS OF ITS ESSENTIAL PURPOSE.

Further information regarding this limited warranty may be obtained by writing:
Telestream, LLC
848 Gold Flat Road
Nevada City, CA 95959 USA

You can call Telestream during U. S. business hours via telephone at (530) 470-1300.

Regulatory Compliance

Electromagnetic Emissions: EN 55032, IEC/EN 61000-3-2, IEC/EN 61000-3-3, FCC Part 15 Subpart B, ICES-003, VCCI 32-1, AS/NZS CISPR 32

Electromagnetic Immunity: EN 550535

Safety: IEC 62368-1, EN 62368-1, CSA C22.2 No. 62368-1-14, UL 62368-1

California Best Management Practices Regulations for Perchlorate Materials:
This Perchlorate warning applies only to products containing CR (Manganese Dioxide) Lithium coin cells. Perchlorate Material-special handling may apply. See www.dtsc.ca.gov/hazardouswaste/perchlorate.

Obtaining Support | Information | Assistance

Contact Telestream for support, information or assistance, as indicated below.

Resource	Contact Information
DIVA Technical Support | Information | Assistance | FAQ | Forums | Upgrades	Web site: telestream.net/telestream-support/DIVA/support.htm Web site: telestream.net/telestream-support/DIVA/support.htm Support Email. support@telestream.net US Phone: 877-257-6245 International Phone: +1 530-470-2036 Support hours: Monday - Friday, 7am - 6pm Pacific time. P1 Support: 24 x 7. Terms and times of support services vary, per the terms of your service contract with Telestream. Depending on problem severity, Telestream responds within 24 business hours. For P1 issues, Telestream responds within one hour.
Product Licensing	Web site: telestream.net/telestream-support/DIVA/support.htm License Email: license@telestream.net
Telestream, generally	Web site: telestream.net Sales and Marketing Email: info@telestream.net

Resource	Contact Information
Telestream Reseller Support	If you purchased your product from a reseller, please contact your reseller for support.
International Reseller Support	Website: telestream.net See the Telestream website for your regional authorized Telestream reseller.
Telestream Technical Writers	Email: techwriter@telestream.net If you have comments or suggestions about improving this document or other Telestream documents—or if you've discovered an error or omission, please email us.

DIVA Library

For more information about DIVA, browse the DIVA library at telestream.net/telestream-support/DIVA/support.htm.

Contents

DIVA Web App 24

About DIVA Web App 24
Launching DIVA Web App and Connecting to DIVA 26
Setting Your DIVA Web App Preferences 27
    Changing the Default Home View 27

Menus, Panes, and Buttons 28

Table Control Buttons 28
Top Banner 29
    Enabling or Disabling Banner Shortcut Buttons 29
    Search Button 29
    Jobs Button and JOBS View 29
        Step Column 30
        Mounting Step 30
        Dismounting Step 30
        Positioning Step 30
        Reading Step 30
        Writing Step 30
        Deleting Step 31
        Transferring Step 31
        Waiting for Operator Step 31
        Waiting for Resources Step 31
        Canceling a Job 31
        Job Priority 31
        Job Resubmission 32
    Job History Button 32
    SPM Actions Button 32
        Viewing and Controlling SPM Actions 32
    Migration Tasks Button 33
    System Health Status Button 33
        Refreshing the System Health Status Display 33
    Notify Button 33
DIVA Dashboard 34
    System Info 34
    Utilization Stats 34
    Alerts 34
    Storage Resource 35
Side Navigation Bar 35

Content Management Button 35
Jobs Tab 35
Job History Tab 35
Displaying the Job History for an Object 35
Displaying Catalog Commands as Buttons 35
Displaying Object Properties 36
Catalog Browsing Tab 36
CATALOG BROWSING View 36
Browsing the Catalog 37
Displaying the Context Menu for an Object 37
Object Explorer Button 37
Managing Assets with Object Explorer 37
Archive Submission Button 38
Search Button 38
SPM Actions Button 38
Resource Management Button 38
Unmanaged Storage Tab 39
UNMANAGED STORAGE View 39
Viewing the UNMANAGED STORAGE View 39
Actors Tab 39
Viewing the ACTORS View 39
Media Tab 39
Viewing the MEDIA View 40
Source Media Priority 40
Disks Tab 40
DISKS View 40
Viewing Storage Options 40
Cloud Disks Tab 41
Cloud Disks View 41
Viewing the CLOUD DISKS View 41
Proxy Disks Tab 41
Attachment Disks Tab 41
Libraries Tab 41
The Libraries View 41
Viewing the LIBRARIES View 42
Drives Tab 42
Drives View 42
Viewing the DRIVES View 42
Tapes Tab 42
Viewing the TAPES View 42
Tape Group Clones 43
Cloning Missing Content Exclusively 43
Cloning Tape Groups Manually 43
Cloning Tape Groups Automatically 44
Repacking Tape Automatically or Manually 44
Troubleshooting Button 45
Logged Events Tab 45
Drive Alert Logs Tab 45

Library Alert Logs Tab 46
Analytics Charts Tab 46
Migrations Button 47
Configuration Button 47
User Management Tab 47
Managing User Accounts 47
Adding a New DIVA User 48
DIVA User Roles 48
License 49
License During an Upgrade 49
Editing a User Configuration 49
Deleting a User 50
Configuring Your User Settings 50
LDAP Authentication 50
Resources Tab 50
System Settings View 50
Library Storage View 51
Storage Plans Settings View 55
Cloud Storage View 55
Disk Storage View 56
Proxy Storage View 56
Attachment Storage View 56
General Settings View 56
Services Tab 57
Analytics Tab 57
Database Backup Tab 57
Metadata Tab 57
API Translation Tab 57
Workflow Tab 57
Log Out Button 57
Migrations Menu 57
Start New Migration Task 58
Migration Tasks 58
Local Settings 58

Jobs 59

Setting the Job Files Path Root and Files Parameters for an Archive 59
Archive Job with Delete on Source 60
Launching a Restore Job 63
Multiple Restore Jobs 66
Restoring to Multiple Destinations 66
Copying an Object 66
Copying an Object as a New Object 67
AXF Archive and Restore Jobs 69
Failure Conditions 69
Partial File Restore Jobs 70
Submitting a Partial File Restore Job 73

Delete and Delete Instance Jobs 76
Deleting an Object 76
Stage Jobs 77
Copy Jobs 78
Copy As Jobs 78
Quality of Service 79
Job (Formerly *Request*) Options 79
Server (Source/Destination) Connect Options 80
Media (Array/Group) Archive and Restore Settings 80
Priority of Application of Quality of Service (QOS) 80
Order of Priority 81

Unmanaged Storage 84

Unmanaged Storage Repositories 85
Defining an Unmanaged Storage Repository 85
Common Job Parameters for Unmanaged Storage Repositories 85
General Parameters 87
Files Path Root Parameter 87
Root Path Parameter 87
Style Paths 89
Metasource Parameter 90
Connect Options Parameter 90
Archiving and Restoring Filename and Path Renaming Rules (`-arch_renaming`, `-rest_renaming`, `-arch_path_renaming`, `-rest_path_renaming`) 90
Connection Timeouts Option (`-list_timeout` | `-transfer_timeout` | `-control_timeout`) 93
Deleting Source Server Content after Archiving (-allow_delete_on_source) 94
DIFWAV 94
DIRS_FIRST 94
FILES_FIRST 95
FTP Socket Window Size (`-socket_window_size`) 96
FTP Socket Block Size (`-socket_block_size`) 96
FTP Passive Mode Transfers (`-pasv`) 96
Ignoring Files During Restore 96
Metadata Restore Option (`-rest_metadata`, `-rm`) 97
MSS 97
MSS Server in MXF Mode (`-mxf`) 98
Number of Actors to Retry Option (`-num_actors_to_retry`) 98
OMNEON 98
Quality of Service (`qos=`) 98
Restore Ignore Option (`-rest_ignoring`) 99
Restore Ignore File Relative Paths Option (`-ignore_relative_path`) 99
Restore AXF Mode Option (`-axf`) 100
Restore Specific Order Option (`-file_order`) 100
SEACHANGE 100
Server FTP User Log In (`-login`) 100

Server CIFS User Log In for CIFS and FTP (-user) 100
Server Password (-pass) 101
Server Connection Port (-port) 101
Transcode Format Option (-tr_archive_format,-tr_restore_format) 101
Transcoder Name Option (-tr_names) 101
Using a Temporary Filename when Restoring to a Destination 102
Avid MSS (Program Stream) Servers 103
MSS with Independent Storage 103
MSS with Shared Storage 104
MSS with Shared Storage in MXF Mode 104
MSS with DIVA_archiveVirtualObject 104
Avid Airspace Servers 105
Supported -title and Third-Party IDs 105
Avid Interplay Server 106
Configuring an Avid Interplay Server 106
Adding an Avid Interplay Server 106
Restoring an Avid Interplay Profile 107
SeaChange BMS and BMC Servers 107
SeaChange BML Servers 108
SeaChange BMLe and BMLex Servers 109
PD 110
VIX 110
-ftp or -cifs 111
Leitch vR Series Servers 112
Leitch Nexio Servers 112
Grass Valley Profile Servers 113
Grass Valley UIM Gateway 114
Grass Valley K2 Servers 116
Grass Valley M-Series iVDR Servers 117
Sony Optical Drives 118
Sony Hyper Agent Servers 118
Sony MAV70 Servers 119
Omneon Spectrum MediaDirector Servers (QuickTime) 120
Omneon MediaGrid Content Storage System 122
Quantel Power Portal Gateway 123
Standard FTP and SFTP Servers 125
Local Source Servers 126
Disk and CIFS Source Servers 127
Configuring Cloud Object Storage as Unmanaged Servers (Azure, S3...) 128
Expedat Servers from Data Expedition 129
Metasources 131

Media Storage Formats 137

Storage-Media Formats 137
Tape Storage-Media Format 137
Disk Storage-Media Format 138

Object Instances Media Format 138
AXF Objects 138
Native File and Folder Support 139
DIVA Complex Objects 140
DIVA Connect Complex Object WAN Transfers 141
Example 141

Data Management on Managed Disks 142

Arrays, Disks, and Cache 142
Creating a Disk Array 143
Adding a Disk to a Disk Array 144
Adding an Actor to a Disk 145
Notifying the Manager and SPM 145

Data Management on Object Storage 146

Cloud Storage Setup 146
Adding a Cloud Account 146
Creating a Cloud Bucket 146
Connecting Actors to Cloud Buckets 147
Target Media Archives 147
Supported Object Storage implementations 147
S3-Based Object Storage (Amazon S3 Integration) 148
Amazon S3 Bucket Limitations 148
Cloudian 148
EMC Elastic Cloud Storage (ECS) Integration 149
Alibaba OSS 149
NetApp StorageGrid 149
Scality Zenko 149
S3 Server-side Encryption 149
SSE-S3 or SSE-KMS Bucket Policy 150
Google Cloud Storage 150
Creating a Service Account 151
Configuring DIVA for Google Cloud Storage 152
Microsoft Azure Blob Storage 152
Oracle Cloud Infrastructure (OCI) 153
File Encryption Status 153
DIVA Web App Support 153
Amazon S3 Transfers 153
Archiving to an S3 Disk 153
Restoring from an S3 Disk 154
Oracle Storage Cloud Transfers 154
Object Storage Destinations 154
EMC ECS Object Store Integration 155

Content Auto-Indexing 156

Supported Storage Type and Functionality 156
Object Storage Auto-Indexing 156
Cloud Array Parameters for Object Auto-Indexing 158
Auto-Index and AXF Archive Locations 159
Clearing Files from Failed Auto-Indexing 160
Disk Auto-Indexing 160
Supported Functionalities 161
Configuration 161
Copy and Restore Request 161
Starting and Stopping the Scanning Service 162
Filter with Regular Expressions 163
File-system Action Handling 164
On File Added 164
On File Modified 164
On File Deleted 164
On File Renamed 164
On Sub-directory Renamed 164
Duplicates Handling 165
Instance Delete Behavior 165

AXF Discovery 166

About AXF Discovery 166
Supported Functionalities 166
Scanning Service Start and Stop 167
Scan Events 167
Scan Statistics 168
Rescan of Existing Objects and Instances 168
Rescan of Bad or Incomplete Instances 168
Automatic Scan Restart 169
Actor-Disk Connection Access Type 169

Data Management on Removable Media 170

Tape Groups and Sets 170
Viewing Tape Groups 171
Robot Manager Configuration 172
Configuration Overview 173
Common Robot Manager Parameters 174
Direct Attached SCSI Medium Changer Settings 176
Common Settings for SCSI-based Medium Changer 176
Fiber Channel HBA (Host Bus Adapter) and SCSI Persistent Binding 178
Sony ODA Drives 179
DIVA Web App Settings and Information 181
ACSLS Attached Libraries 182
LibAttach 182

The LibAttach Connectivity to ACSLS 182
Firewall Support 182
robotmanager.conf Common Options 183
Sony PetaServe Library Configuration 184
robotmanager.conf Common Options 184
Alto Disk Archive Library Configuration 187
Configuring Robot Manager 188
Actor AltoVirtualDrives Configuration 188
Syntax: 188
Example: 189
Simulated Media Changers Configuration (for DIVA Simulators) 189
robotmanager.conf Common Options 189
Robot Manager Library Interface 191
Robot Manager Start, Stop, and Restart 191
Robot Manager Library Control 193
Robot Manager System-Level Configuration 194
Robot Manager Activity Logs 195
Media and Drive Type Configuration 195
SCSI_drive_types and ACSLS_drive_types 196
SCSI_tape_types and ACSLS_tape_types 196
Tape Group Configuration 196
Reasons for Creating Multiple Tape Groups 197
Invalid Reasons for Creating Multiple Tape Groups 198
Tape Drive Reservations 199
Tape Label Management 201
Tape Drive Encryption 202
Enabling Tape Drive Encryption 202
AXF LTFS and TAPE Encryption 203
Obtaining the Encryption Key for an Encrypted Tape 203
Tape Compression 203
Viewing Compression-Enabled Tapes 204
Editing Tape Groups 204
Tape Spanning 204
True Remaining Tape Size and Last Written Position 205
Repacking Tape Jobs 205
Cloned Tape Repack 207
Repacking Tapes Automatically 207
Verifying Tape Jobs 207
Tape Eject/Insert 208
Ejecting Tape Jobs 208
Cloned Tape Eject 209
Protected Mode 209
Inserting a Tape Into a Library 210
Export and Import Functions 211
Tape Export 211
Export Limitations 212
Encrypted Tape Export 213
Export Metadata Parameters 213

Exported Tape Metadata Files 214
Exporting Tapes 215
Cloned Tape Export 216
Bulk Tape Export 216
Tape Import 216
The Import Command 217
Importing Tapes 217
Encrypted Tape Import 221
Cloned Tape Import 221
Bulk Tape Import 222
Example Import 222
Export and Import Tape Jobs 223
Exporting a Tape 224
Export/Import Error Handling and Failure Scenarios 225
Export Failed Error Message 225
Invalid Parameter Error During Export 225
Tape Already Exists Error During Import 225
Unsupported Type Error During Import 226
Import Process Terminated Without Importing 226
Export/Import Through the Java API 226
Adding a Tape Library, Corresponding Drives and Tapes into DIVA 229
Prerequisites 229
Software 229
Hardware 229
Credentials 229
Environment Setup 229
Verifying the Environment 229
Confirm a Robot Manager 230
Setting Up the Remote Library 230
Connecting DIVA to a Remote Library 230
Configuring Devices in Windows 230
Configuring DIVA Robot Manager 230
Editing Configuration Files 230
Installing Robot Manager Service 231
Syncing and Adding the Library in DIVA 231
Adding the Robot 231
Configuring Tapes and Drives 232
Final Configurations and Testing 232
Adjusting iSCSI Settings 232
Testing the Configuration 232
Troubleshooting Tips 232
Common Errors and Fixes 232
Logs and Diagnostics 232
FAQs and Additional Resources 232
Damaged Tape Utility Recovery 233
Tape Scanning for Indexing Objects 233
Tape Scan Job Parameters 233
Submitting an LTFS Scan from the DIVA Web App 234

Viewing Scan Tape Jobs 234
Re-indexing Existing Objects 235

Checksum and Content Verification Support 236

About Checksum 236
Viewing Checksum Values 237
Checksum Support Configuration 237
Global Checksum Parameters 237
Enabling the Checksum Feature 237
Manager: Default Checksum Type 238
Manager: Number of Retries Following Failed Checksum 238
Adjusting Checksum Support for Unmanaged Storage 238
External Checksum Source 238
Checksum Type 239
GC Mode 239
Verification Following Archive (VFA) 239
Following Restore (VFR) Verification 239
Configuring Checksum Support for Arrays and Disks 240
Configuring Checksum Support for Tape Groups 240
Configuring Checksum Support for Actors 240
AXF and TEXT Genuine Checksum Modes 241
AXF Genuine Checksum Mode 241
Requirements 241
AXF Genuine Checksum Mode 241
Configuring DIVA Web App Settings 241
Archiving an Object with Genuine Checksum 242
TEXT Genuine Checksum Mode 242
TEXT Genuine Checksum Mode 242
Configuring DIVA Web App Settings 243
The Root File Path 243
Archiving Objects with Checksum 244

DIVA Content Migration Tool 245

About DIVA Migration Tool 245
Migration Tool Installation 245
Migration Tool licensing 246
Windows Files and Folders 246
Configuration 246
Logging Settings 246
Starting and Stopping the Service 247
Migration Tool 247
Migration Task Events 248
Migrating Sources to Targets (Administrators Only) 248
Migration Tasks View 249
Migration Task Functions and Parameters 249
Migration Task Properties 249

Migration Tasks Actions 250
Migration Task Status 250
Migration Task Parameters 250
Migration Tool Start and Stop 252
Advanced Migration Jobs 252
Tape Migrations 252
Migration Tasks and DIVA Jobs 252
Error Handling and Failure Scenarios 253
Migration Error Handling 253
Migration Failure Scenarios 254

Drop Folder Monitor (DFM) 257

About Drop Folder Monitor (DFM) 257
Supported Functionalities 258
Example 258
Features 260
DIVA Connect Web App Monitor 261
DFM Monitor with Logs 261
Start/Stop the Drop Folder Monitor Service 262
DFM Configuration 262
DFM Multiple File Archive 265
Archive Request Failover 266
Replacing an Existing Object with a Dropped Object 266
DFM Restart 267

Drop Folder Manager (Legacy) 268

About Drop Folder Manager (Legacy) 268
Features 269
Single File Mode Drop Folders 269
File Set Drop Folders 270
Common Drop Folder Manager Solutions 270
Basic Solution for Single Mode Folders 270
Basic Solution for File Set Mode Folders 271
Basic Solution for Delete Mode 271
Basic Solution for DIVA Connect 272
Drop Folder Manager Component 272
Administration, Operations and Monitoring 273
Drop Folder Manager Overview 274
Drop Folder Manager Service Start and Stop 274
Drop Folder Manager Service 274
Drop Folder Manager Service Stop 275
Summary of Drop Folder Manager Administration 275
Command-Line Interface 276
Operations and Workflows 277
Algorithm and Workflow for Single File Mode Drop Folders 278
Algorithm and Workflow for File Set Mode Drop Folders 279

Drop Folder Manager 281
DIVA Connect Web App Monitor 281
DFM Monitor with Logs 282
Solutions and Basic Configurations 282
Basic Single Mode Folder Solution 282
Configuring the Basic Single Mode Folder Solution 283
Solution for File Set Folders on Local Disk and FTP Servers 284
Recursive Archive Configuration 286
Configuring the File Set Folders Solution 287
Delete Mode Solution 288
Configuring the Delete Mode Solution 288
DIVA Connect Solution 289
Configuring the DIVA Connect Solution 290
Multiple DFM Instances on the Same Computer 291
Service Configuration 292
Service Update 292
Service Installation 292
Service Removal 292
Service Start 292
Service Stop 292
Service Logging 292
Service Monitoring 293
Additional Configuration Options 293
Additional DFM Configuration 293
Incomplete File Processing: <incompleteProcessingStrategy> 293
MDF Configuration Priority: <mdfConfigPriority> 294
Additional Operations Drop Folder Manager Configuration 294
Delete Before Archive: <deleteBeforeArchive> 294
Number of Files to Archive Wildcard: <numberFilesToArchiveWildcard> 294
Object Name Conversion Rules: <objectNameConvertRules> 295
Original Server: <originalServer> 296
Original Path: <originalPath> 296
Media Name: <divaMediaName> and <divaMediaNamePattern> 296
Additional Monitored Folder Configuration 297
MDF File Extension: <mdfExtension> 297
Server Path: <sourceDestinationDIVAPath> 297
Sample Server Paths for Different Configurations 297
Archive File Path Template: <archiveFilePathTemplate> 299
Archive File Name Template: <archiveFileNameTemplate> 299
File Filter: <fileFilter> 300
Delete Parent and Content Directories: <deleteParentDirectoryAndContentDirectories> 301
Folder Configuration Examples 301
Example of a Local Folder in File Set Mode 301
Example of an FTP Folder in Single File Mode 302
Example of a CIFS Folder in File Set Mode 303
Service Log Configuration 304

Trace Log Configuration 305
Advance DFM Configuration 306
Full DFM Configuration File 306
Global DFM Configuration 306
Folders Default Configuration 308
Folder Specific Configuration 310
File Root Path and Single File Mode Advanced Configuration 312
Configuration Files and Examples 314
Object Configuration 314
Metadata File 315
Sample of the sample.mdf File in XML Format 315
Sample of the sample.mdf File in Original MDF Format 316
Full Drop Folder Manager Configuration File (dfm.conf.ini) 317

Proxy Management 318

About Proxy Management 318
Installing Proxy Service 318
Importing a Proxy 319
Creating a Proxy CSV Manually 320
Exporting a DIVAView Proxy CSV 320
Proxy Import 320
Preparation for Importing Proxies 320
Configuring Proxy Import 321
Configuring Workflow Service Module (WFS) 324
Playback Hosting for Proxies and Images 325
Workflow Initiator 325

Storage Policy Manager 326

About Storage Policy Manager 326
The Object Life Cycle in DIVA 326
SPM Slot Types 327
Actions, Action Steps, and Action States 329
Watermarks and Arrays 330
Best Practices for Watermarking 331
Storage Policy Manager Workflows 334
Storage Policy Manager Tasks 334
Simple Object Life Cycle Example 335
SPM Failed Action Retry and Issue Resolution 335
Action Slot Workflow 337
Storage Slot Workflow 337
SPM Best Practices 338
SPM Configuration 338
General Watermarking Rules 339
Simple SPM Configuration Example 340
Additional SPM Configuration Examples 341
Installation and Configuration 342

Installation Prerequisites 342
SPM Configuration File Parameters 343
Request Type Distribution 347
Storage Plan Definition 349
Storage Plan Settings 349
Recommended Best Practices 349
Creating the Storage Plan 350
Creating Filters 350
Assigning Storage Plans to an Object 352
Slots 353
Configuring Storage Slots 353
Configuring Transcode Archived Slots 359
Configuring Metadata Archive Slots 365
Configuring Restore Slots 369
SPM Delete Object Behavior 374
Storage Plans 376
Adding a Storage Slot 377
Slot Disablement or Deletion 377
Slot Addition 377
Watermark-Based Disk Cleaning Management 378
High Watermark 378
Low Watermark 379
External Storage Plans 379
Validating Configuration 379
Operations 380
SPM Service Command Line Management 380
SPM Information in the DIVA Web App 381
SPM Actions View Filter 381
Assigning a New Storage Plan 382
Rescheduling Failed Actions 382
Failed Action Completion 383
Storage Policy Manager Action Result Codes 383
SPM Logging Feature 386

AVID Connectivity and Avid for DIVA Applications 387

About Avid Connectivity 387
Avid Interplay (Avid PAM, Avid MediaCentral Production) Supported Environments 388
AvidForDIVArchive and DIVA Release Compatibility 388
AM Communicator (AMC) 388
Direct ISIS Connectivity AM Communicator 388
Avid Workflows for DIVA (AWD) 389
Installing Avid Connectivity and AvidForDIVArchive 390
AMC Installation, Configuration, and Operations 390
Installation Overview 390
Installing AM Communicator 391
Command Line Options 391

Configuring AMC for Direct ISIS/Nexis Connectivity 393
Installing the ISIS Client 393
AM Communicator Server Configuration Parameters 394
Configuring Avid Archive Manager 394
Enabling Best Effort Restore 396
Configuring Avid Interplay to Store AAF Files 396
Operations 397
Archiving 397
Restoring 397
Deleting 398
Workflows 398
Direct Connect AMC Workflows 398
AM Communicator Configuration Parameters 400
AWD Installation, Configuration, and Operations 404
Installing Avid Workflow for DIVA (AWD) 404
Configuration 404
Operations 408
The Interplay Profiles 408
The Job Folders 408
Avid Archive Manager Updater Tool 409
System Requirements 409
Installing and Configuring the AM Updater Tool 409
Usage 412

Frequently Asked Questions 413

DIVA Operations 413
Export / Import Operations 414
Analytics 415

DIVA Glossary 416

DIVA Web App

You can monitor, control, and supervise DIVA operations via the DIVA Web App.

Topics

- About DIVA Web App
- Launching DIVA Web App and Connecting to DIVA
- Setting Your DIVA Web App Preferences

About DIVA Web App

DIVA is a content-management middle-ware and a media-asset manager. DIVAstands for *Digital Intelligent Versatile Archive*.

You can use the DIVA Web App from any computer that has TCP/IP connectivity to the DIVA Core server.

DIVA provides the following functionalities:

- Monitoring jobs that have been submitted either through the DIVA REST API or through the DIVA Web App
- Monitoring the status of actors, drives, and disks connected to DIVA
- Initiating commands, such as Archive, Restore, Partial File Restore; and submitting them for execution
- Managing tapes for libraries controlled by DIVA, such as internalizing, externalizing, and tape repacking
- Browse or search for, and manage,. your content
- View proxies, edit metadata, create markers and segments, and upload attach-ments to Objects

The following table describes various DIVA use cases.

Use Case	How It Works	Related Features
Protect my data	Once my content is archived, it is replicated on different storage media (tape, disk, object storage). My content can be replicated to a disaster recovery site.	• Storage Plan Manager • DIVA Connect
Migrate my content to different storage technologies		• Migration service
Access my content when I need it		• Restore • Partial restore • Multiple-restore
Organize my content		• Object browser
Control my archive and be aware of its content	Browse the content on the archive, make searches, preview the content.	
Collaborate	Exchange objects between sites using the network or by shipping media .	• Diva connect • Tape import/export • LTFS tape scan
Use the archived objects to enrich my workflows.		

The following table describes how to get your DIVA-managed content in various use cases:

Use Case	Method
I already have a MAM or another DIVA automation.	The archive jobs are supplied by the automation.
I don't have a MAM or DIVA automation.	Use DFM, bucket or disk auto-indexing.
I don't need archive storage or data preservation, but I need my assets to be managed by DIVA.	Use bucket or disk auto-indexing.
I have some LTFS tapes containing media files.	Use LTFS tape-scan functionality.
I want to archive something via DIVA Web App.	The object explorer offers access to an upload-archive function.

Note: Telestream recommends using the Chrome browser to access the DIVA Web App.

Note: Use the DIVA Web App to select most of the DIVA configuration settings and options.
For intensive archive operations, use a broadcast-automation or media-asset-management (MAM) system, rather than the DIVA Web App.

Note: In this guide, references to the name of a view in the user interface use the same spelling and capitalization as the name of the view; italicized to indicate the word is the name of the view. For example, to search for the *JOBS* view by name, you can enter capital letters; for example, "JOBS."

Caution: Incorrect or incomplete changes in the DIVA Web App can adversely affect DIVA or Manager operations.
Be sure to assign the appropriate roles to users so that they have access to the correct subset of features for their roles. See *DIVA User Roles*.

Launching DIVA Web App and Connecting to DIVA

To launch the DIVA Web App and connect to DIVA, do the following:

1. Launch Chrome.
2. Browse to https://localhost:8765.

   After the user interface loads, DIVA displays the login window.
3. To log onto the DIVA server, enter your DIVA user credentials, or, if you have LDAP enabled, your LDAP credentials.

   See *LDAP Authentication*.

   After you log in, DIVA displays the home page, which by default is the Dashboard.

Note: You can change the default home page to the Jobs view, or--if you're licensed for Analytics--to the Analytics dashboard. See *Setting Your DIVA Web App Preferences*.

Note: If you're accessing DIVA from a workstation, or if you're using a distributed DIVA system, substitute the keyword *localhost* in the URL with the domain name or IP address of the Manager server.

Note: If the DIVA Web App can't establish a connection to DIVA, it displays a 404 error instead of the DIVA Web App user interface. The DIVA Web App user interface app is served by the web server.

Setting Your DIVA Web App Preferences

To specify your preferences, do the following:

1. Browse to *Configuration* > *User Management* > *User Settings*.
2. Set preferences, including date and time display preferences, home view selection, and banner shortcuts selection.

DIVA stores all user settings in the database on a per-user basis. If you log into the DIVA Web App as the same user on another machine, your settings travel with you.

Changing the Default Home View

To change the default home view from System Dashboard to Jobs, do the following:

1. Browse to *Configuration* > *Local Settings* > *Customization*.
2. Choose the *Customization* settings desired.
3. Click *Apply Changes*.

Note: For a list of changes made to DIVA for this release, see the *DIVA Release Notes* for this release.

Menus, Panes, and Buttons

You can monitor, control, and supervise DIVA operations via the DIVA Web App.

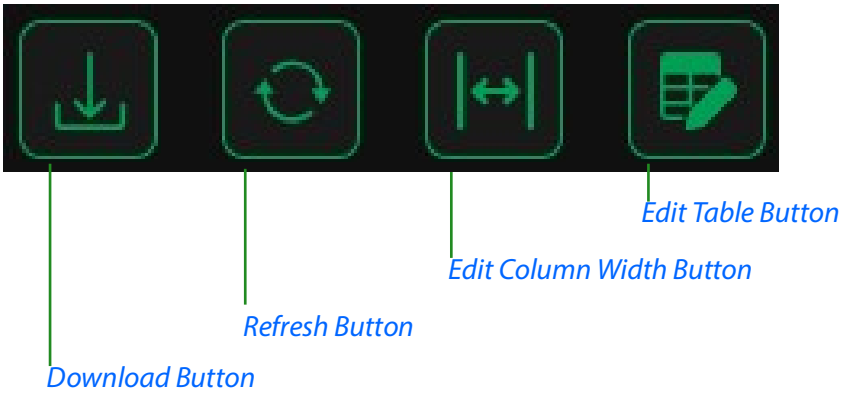
Topics

- Table Control Buttons
- Top Banner
- DIVA Dashboard
- Side Navigation Bar
- Content Management Button
- Resource Management Button
- Troubleshooting Button
- Migrations Button
- Configuration Button
- Log Out Button
- Migrations Menu

This chapter describes the functions of the menus, panes, and controls on the DIVA Web App.

Table Control Buttons

Table control buttons give you control over the display and export of many tables in DIVA. Descriptions of the table control buttons follow.



Download Button. Downloads the table into a CSV file.

Refresh Button. Refreshes the data displayed in the table

Edit Column Width Button. Allows you to edit the table column widths.

Edit Table Button. Allows you to edit the table design.

Top Banner

In the top banner, DIVA displays shortcut buttons for *Jobs*, *Job History*, *SPM Actions*, *Catalog Browsing*, and *Archive Submission*.

Enabling or Disabling Banner Shortcut Buttons

To enable or disable the banner shortcut buttons, do the following:

1. Browse to *Configuration* > *User Settings* > *Customization*.
2. Under the *Shortcuts* heading, toggle the settings for each shortcut button as desired.

Search Button

When clicked, the *Search* button opens the Search field.

Note: You can perform a global search of your content from the *Search* field. This searches across object name, collection, defined metadata fields, segment markers, and uploaded attachments.

Jobs Button and JOBS View

When clicked, the *Jobs* button opens the *JOBS* view.

The *JOBS* view displays the jobs that DIVA is executing, the jobs that are pending, and the jobs that were completed, were canceled, or that triggered warnings during execution. This feature works only when the DIVA Web App is connected to DIVA. DIVA doesn't display jobs completed or canceled before this connection is made.

Once a job is complete the job remains in the *JOBS* view, by default, for 30 seconds. After that time limit, DIVA moves the jobs to *JOBS HISTORY*. In the user settings, you can set the amount of time a completed job remains in the *JOBS* view.

Clicking a job produces an informational view about the selected job.

After a particular job ID is determined, you can locate it in the *JOBS HISTORY* view. You can search for it in the displayed jobs using the Filter button at the top of the window.

A yellow triangle in the State column indicates the job completed but had a warning associated with it. A red triangle in the State column indicated the job failed with an error. Click the job ID to open the *Job History* > *Properties* view to identify any errors or failure events. On the Properties view the cause of the failure is stated in the Abort Reason area on the right, and also in the Job Events area at the bottom of the view.

Step Column

The Step column indicates the current operation of the job being performed by DIVA. A short description of each step is as follows:

Mounting Step

The mounting step is completed after the tape is fully threaded, positioned at the tape header, and the DIVA label on tape verified with that of the bar code label.

If the label doesn't match, the job is aborted, and the tape is set to Not Writable. This situation could occur if there is a mismatch in the Actor-Drives configuration, or the tape has already been used in another archive system (and therefore has a tape label not usable by DIVA). The latter example is a protection feature in shared library environments, where the tape has not been set to Set ID 99 (that is, not in use by DIVA).

Tapes from other archive systems must first have their tape label cleared before using them in DIVA. Contact Telestream.

Dismounting Step

This step involves ejecting a tape from a drive. An Actor first issues an Eject command to the drive (in which it rewinds, un-threads and ejects the tape), and the Robot Manager issues a dismount command to the library to return it to a tape bin. If the drive can't complete the job, the job is terminated and the drive set to Offline.

Positioning Step

When reading from tape, the tape is positioned to the selected object. When writing to tape, the tape is positioned to the End of Data (that is, to the position where the last object was written). If this process takes too long, DIVA times out the operation and attempt to dismount the tape. If this also fails, the drive is set to Offline.

Reading Step

The object displayed in the Object Name column is read from tape. If this step takes too long (for example, the drive is in a hung state), DIVA times out the step and attempt to use another drive (or instance if available).

Writing Step

The object displayed in the Object Name column is being written to tape. If this step takes too long, (for example, the drive is in a hung state), DIVA attempts to write the object to another tape in another drive.

Deleting Step

This step is rewriting the tape's label and moving the End of Data pointer therein before writing to the tape in an archive operation. This is seen on a tape which has previously been used by DIVA but has since had all objects deleted and consequently returned to the Unused Tapes Sets.

Transferring Step

Data is transferred either to a source server from an actor cache, or from a destination server to the actor cache.

Waiting for Operator Step

This step holds the job in a suspended state and is waiting for human intervention, such as inserting tapes in library's Cartridge Access Port.

Waiting for Resources Step

DIVA resources required for this job are currently in use by another job and the job is executed when they become available. Resource availability can also be influenced by the Priority of other jobs much lower in the queue.

DIVA incorporates an intelligent feature (which must be enabled in the DIVA configuration) whereby jobs with a lower priority are boosted over higher priority jobs if they involve a tape that is already mounted from a previous job. This feature can substantially reduce the amount of tape mounting and dismounting, and speed up the executions of all jobs overall.

Canceling a Job

To cancel a job that is running or pending, do the following:

1. Select the specific job to be canceled.
2. Click *Cancel* from the command menu or from the *Current Jobs* context menu.

Note: The current operation (or step) on a job that is currently being executed may need to complete before the job is actually canceled by DIVA.

Job Priority

If there are several pending jobs in the Current Jobs Queue, DIVA processes each job based on its priority.

If a specific job needs to be executed before (or after) the jobs that precede it in the queue, the job's priority can be manually adjusted to be higher (or lower) than that of the preceding jobs. You can also raise or lower a job's priority via the DIVA REST API, using a third party archive initiator.

Increasing a pending job's priority doesn't stop, or pause, any jobs that are currently executing. This simply changes the order in which the pending job is processed by DIVA, except if a resource being used by a running job (such as a specific tape) becomes available after that running job has completed an operation.

Note: If you've enabled the `DIVA_CORE_PRIORITY_TIER` setting in the DIVA configuration, a priority-tier job that involves a tape already mounted may be executed before jobs higher in the queue.

By default, DIVA periodically increments the priority of all jobs in the queue. This prevents low priority jobs (such as Copy to Group) from continually being overridden by higher priority jobs and getting stuck indefinitely in the queue.

Job Resubmission

A previously completed or failed job can be resubmitted to DIVA using the Retry command. This is useful for resubmitting similar jobs where few of the details change between them. For a job that was terminated (for example, because a parameter was incorrectly entered, or a server was briefly offline), the failed job can be retried without having to submit a completely new job.

Job History Button

When clicked, the *Job History* button opens the *JOB HISTORY* view.

The *JOB HISTORY* view displays completed jobs. Users can retry a job from this view, and in doing so the original job form will be presented pre-populated with the original parameters of the job. You can change these values before re-submitting the job.

SPM Actions Button

When clicked, the SPM Actions button opens the *SPM ACTIONS* view.

The *SPM ACTIONS* view extracts information from the DIVA Core database about actions initiated to DIVA by the SPM module.

Viewing and Controlling SPM Actions

To view and control SPM actions, do the following:

1. In the left menu, browse to *Content Management* > *SPM Actions*.
2. (Optional) To limit the type and number of results returned from the query, click the *Filter* button at the top of the view.
3. To run the query, on the *Filters* popup dialog, click *Apply Filters*.
4. On a result returned by the SPM Actions, under *Actions*, click the *SPM Actions Context* menu (ellipses).
5. Select from these options:

- *Reschedule Action*—If the SPM-initiated job failed (for example, the medium or associated Actors for the slot were unavailable), select to retry this SPM Action.
- *Mark Action Completed*—Select to mark the SPM action as complete, if it's in another state, such as Partially Completed.

Migration Tasks Button

When clicked, the *Migration Tasks* button opens the *MIGRATION TASKS* view.

System Health Status Button

When clicked, the *System Health Status* button opens the *SYSTEM HEALTH STATUS* menu.

From here, you can view the status of components in the system, including such details as License, Search Service, Actors, Disks, Libraries, Drives, Workflow Service, and tape groups. Green signals good, yellow signals a warning, and red signals an error.

If you enable, in user settings, the monitoring of the back-end microservices, these also appear.

When a user logs into the DIVA Web App, it automatically connects and checks the system status. Contact Telestream Support if you still can't connect after attempting to resolve the error.

Disconnect the DIVA Web App when not in use by logging out of the app and closing the browser being used.

Note: `SYSTEM HEALTH STATUS` components are updated live when the DIVA Web App receives a notification from the back end.
`SYSTEM SERVICES STATUS` components are updated every polling period. The default value is 60 seconds.

Refreshing the System Health Status Display

To refresh the system health display, press the refresh button next to `SYSTEM HEALTH STATUS`.

`SYSTEM SERVICES STATUS` forces a live update for the state of the components, fetched from the back end.

Notify Button

When clicked, the *Notify* button opens the *NOTIFY* menu. The button informs the user of notable changes that have been made. The *NOTIFY* menu includes the following commands:

Notify Manager. When clicked, DIVA opens the *NOTIFY MANAGER* dialog. This allows DIVA to notify Manager service of any relevant configuration changes.

Notify SPM. When clicked, DIVA opens the *NOTIFY SPM* dialog. This allows DIVA to notify the SPM service of any relevant configuration changes.

Rebuild Search Indexes. When clicked, DIVA opens the *REBUILD INDEXES* dialog, and requests a rebuild of the search indexes. Rebuilding search indexes might be necessary if the metadata configuration has changed.

Note: To notify the actors of any changes in the actor configuration in the DIVA Web App, the actors must be running and connected to the Manager to receive the notifications.

See *Notifying the Manager and SPM*.

DIVA Dashboard

DIVA displays system data and system-usage data in the following four panes:

- *System Info*
- *Utilization Stats*
- *Alerts*
- *Storage Resource*

  Displays storage in use. You can determine whether DIVA displays or doesn't display this. See *Configuring Your User Settings*.

System Info

The *System Info* pane displays key properties from the license: customer name, system ID, and expiration date. It also displays the software version of DIVA, and how long the Manager has been running.

Utilization Stats

The *Utilization Stats* pane displays the total space available for storage, and how much of that space is currently consumed. There is a combined view of Cloud, Disk, and Tape metrics in their individual categories.

Alerts

The *Alerts* pane displays a summary of the metrics, organized by info, warnings, and criticality. The metrics provide a quick view as to whether all drives, libraries, disks, cloud disks, and actors are available.

Storage Resource

The *Storage Resource* pane displays utilization statistics by individual resource.

Side Navigation Bar

The navigation bar on the left side of the DIVA Web App user interface displays the following buttons:

- *Content Management Button*
- *Resource Management Button*
- *Troubleshooting Button*
- *Migrations Button*
- *Configuration Button*
- *Log Out Button*

Content Management Button

When clicked, the *Content Management* button opens the *Content Management* menu, displaying a number of tabs.

When you click any of the following tabs, DIVA displays the content indicated.

Jobs Tab

When clicked, DIVA opens the *JOBS* view.

See *Jobs Button and JOBS View*.

Job History Tab

When clicked, DIVA opens the *JOB HISTORY* view.

Displaying the Job History for an Object

To display the job history view for an object, do the following:

1. Select the object.
2. Click *Content Management* > *Job History*.
3. (Optional) To display the job information for the selected object, click the three dots next to an object, or the object ID.

Displaying Catalog Commands as Buttons

To display catalog commands as buttons at the bottom of the window, select the check box next to an object.

When you select a checkbox next to an item in a table, and you select multiple items, DIVA displays actions at the bottom of the table. DIVA performs any of these actions on all the selected items.

Similar functionality is available from the actions menu in each row. These actions affect only the single item. The multi-select actions appear in a menu under the table. You can select many actions from this menu.

Displaying Object Properties

To display an object's properties, instances, and components, click the object name.

Catalog Browsing Tab

When clicked, DIVA opens the *CATALOG BROWSING view*.

CATALOG BROWSING View

The CATALOG BROWSING view displays the objects in DIVA. You can perform the following actions on objects:

- Copy
- Copy To New
- Stage
- Restore Object
- Partial Restore
- Multiple Restore
- Delete Object
- View Object Properties

Also, you can search objects using the Filter button .

DIVA doesn't automatically specify an instance of the object. You can manually select a specific job instance from the drop-down menu of the Instance field. If the Instance field is left blank, DIVA applies the command to all instances of the object.

For example, if you select Delete without specifying an instance number in the job, DIVA deletes all instances of the object. With a particular instance specified, DIVA deletes only that instance. This includes removing the instance number entirely from the job.

You can't delete all instances of an object from this view by leaving the Instance field empty. This view doesn't permit you to delete the last instance of the object. DIVA automatically terminates this job.

If a file (or part of the file) of an object instance is spanned across two or more tapes, and only one tape of the set is externalized, the instance is still considered externalized.

However, an object (that is, all instances) is considered externalized only if all instances of that object are externalized.

Browsing the Catalog

To browse the catalog, in the left menu, click *Content Management* > *Catalog Browsing*.

Displaying the Context Menu for an Object

To display the context menu for an object, select an object and click the *More* menu.

Object Explorer Button

When clicked, DIVA opens the *OBJECT EXPLORER* view.

The *OBJECT EXPLORER* view enables you to manage a large volume of media assets. In the *OBJECT EXPLORER* view you can do the following:

- Create, rename, and organize folders and files
- Search for, preview, and access video files using metadata, such as creation date, file type, or tags
- Set permissions for accessing sensitive file
- Seamlessly export and import media files to and from editing software
- View and manage file versions
- Tag files with custom tags in addition to system-defined metadata
- Monitor and log user activity
- Extract reports on file usage and access patterns
- Manage project-based access of team members

Managing Assets with Object Explorer

To manage media assets with Object Explorer, do the following:

1. In the DIVA Web App, browse to Content Management > Object Explorer.

   The DIVA Web App displays the *OBJECT EXPLORER* view.

   See *Object Explorer Button*.
2. Select the type of asset desired from the navigation bar on the left side.

3. Change the display in any of the following ways:
   - To refresh the display, click the *Refresh* button.
   - To edit the column width, click the *Edit Column Width* button. Drag the side of any column to increase or decrease its width, and click the *Edit Column Width* button again.
   - To edit the table, click the *Edit Table* button. Select the names of columns you want displayed, and deselect the names of columns you don't want displayed, and click the *Edit Table* button again.
4. To download one or more assets, check the box to the left of the names of the desired assets; and click the *Download* button.
5. To move objects, copy objects, edit object tags, or download objects, select the desired action from the *Action* drop-down menu; and click *Update Objects*.

Archive Submission Button

When clicked, DIVA opens the *ARCHIVE OBJECT* view.

The *ARCHIVE OBJECT* view enables you to submit objects for an Archive Job.

Search Button

When clicked, DIVA opens the *SEARCH* view.

The *SEARCH* view enables you to search for DIVA elements.

Note: You can perform a global search of your content from the *Search* field. This searches across object name, collection, defined metadata fields, segment markers, and uploaded attachments.

SPM Actions Button

When clicked, DIVA opens the *SPM ACTIONS* view.

The *SPM ACTIONS* view lists SPM actions.

Resource Management Button

When clicked, the *Resource Management* button opens the *Resource Management* menu, displaying a number of tabs.

Click any of these tabs to display the content indicated.

Unmanaged Storage Tab

When clicked, DIVA opens the *UNMANAGED STORAGE* view.

UNMANAGED STORAGE View

The *UNMANAGED STORAGE* view provides information about the sources and destinations identified in DIVA. This view displays the source or destination server Name, Product System, Type, Address, and First Utilization Date.

Viewing the UNMANAGED STORAGE View

To view the *UNMANAGED STORAGE* view, do the following:

1. Browse to *Resources Management* > *Unmanaged Storage*.
2. (Optional) To display the *PROPERTIES* view for a resource, click one of the entries in the *Name* column.

   The information on the *PROPERTIES* view is not editable. It's only for informational purposes. However, you can copy certain settings, such as Root Path or Options.

Actors Tab

When clicked, DIVA opens the *ACTORS* view.

Viewing the ACTORS View

The *ACTORS* view provides an indication of the status of each Actor defined in the DIVA Web App and any current jobs.

To view the *ACTORS* view, do the following:

1. To open the *ACTORS* view, do one of the following:

- Browse to *Resources Management* > *Actors*.
- On the Dashboard, select *Actors*.

  The top right of the *ACTORS* view includes a button to toggle between the table view and the item view. The last view used is the initial view shown on login.

2. To display the currently running job on an Actor, select one of the Actors.

   If the Manager can't establish a connection to an Actor, DIVA displays the Offline status.
3. To display the Actor properties view, click an Actor Panel, or access the menu from the vertical-dot button.

Media Tab

When clicked, DIVA opens the *MEDIA* view.

Viewing the MEDIA View

This view displays information for each of the tape groups and disk arrays identified in the DIVA system. You can filter the search with the Filter button at the top of the view.

To view the MEDIA view, do the following:

1. Browse to *Resources Management* > *Media*.
2. Click the Filter button .

   DIVA opens a filter dialog.
3. (Optional) To search for a name, enter a term into the *Name* field of the Filter dialog. The name can be a full or partial media name including wildcards. Use an asterisk to display all media names.
4. (Optional) To search for a type, enter a term into the *Type* field of the Filter dialog to select viewing all media types, only *Tape Groups*, or only *Arrays*.
5. To display the *Media Properties* view with details about that tape group or array, click on the tape group or disk array.

   The view is for informational purposes only and is not editable.

Source Media Priority

The Source Media Priority determines which source instance is preferred (according to the media where the instance resides) during the instance selection process of a Restore, Partial File Restore, and Copy To Group job. Instances on media with a higher priority are preferred. Cloud instances are only copied or restored if all local instances are offline, or no local instances exist. This is an absolute condition independent of the Source Media Priority.

Disks Tab

When clicked, DIVA opens the *DISKS* view.

DISKS View

The *DISKS* view displays the online status and capacities of disks configured in DIVA. You can set the status of a disk through the *DISKS* view using the three dots menu and selecting the appropriate status. If DIVA has automatically set a disk to Offline, you must investigate the cause of the error before setting the disk back to Online. When DIVA encounters an I/O error with the disk, DIVA automatically sets the disk to Offline.

The column titled Consumed Size represents the space consumed by the content on disk.

Viewing Storage Options

To view storage options, do the following:

1. Browse to *Resources Management* > *Disks*.

2. To display the *DISK PROPERTIES* view, click a disk name.
3. To display a panel showing the files contained on the disk, under the disk properties, click *Files on Disk*.

Cloud Disks Tab

When clicked, DIVA opens the *CLOUD DISKS* view.

Cloud Disks View

The *CLOUD DISKS* view displays the online status and capacities of the cloud disks configured in DIVA. The status of a cloud disk can be set through the *CLOUD DISKS* view by clicking the three dots next to the Disk Name and selecting the appropriate status. If DIVA has automatically set a disk to Offline, the cause of the error must be investigated before setting the disk back to Online. If DIVA encounters an I/O error with the disk, DIVA automatically sets the disk to Offline.

The column titled *Consumed Size* represents the space consumed by the content on disk. This column is especially useful for cloud accounts with unlimited disk space, because it provides visibility into the amount of content stored in the cloud.

Note: A disk linked to an OCI storage account will always report a Consumed Size of 0.

Viewing the CLOUD DISKS View

To view the *CLOUD DISKS* view, browse to *Resources Management* > *Cloud Disks*.

Proxy Disks Tab

When clicked, DIVA opens the *PROXY DISKS* view. You can use the filter to find proxy disks. The *PROXY DISKS* view displays a table of data about your proxy disks.

Attachment Disks Tab

When clicked, DIVA opens the *ATTACHMENT DISKS* view. You can use the filter to find attachment disks.

Libraries Tab

When clicked, DIVA opens the *LIBRARIES* view.

The Libraries View

The *LIBRARIES* view displays basic information and status for each of the libraries connected to DIVA. The *LIBRARIES* view displays the Serial Number, Name, Type, ACS,

Status, First Utilization Date, Total Tapes, Total Data Stored, Total Capacity, and Free Capacity for the connected Libraries.

Viewing the LIBRARIES View

To view the LIBRARIES view, do the following:

1. Browse to *Resources Management* > *Libraries*.
2. To view properties, Click the three dots in the library area.

   This view offers information concerning the specified library.

Drives Tab

When clicked, DIVA opens the *DRIVES* view.

Drives View

The *DRIVES* view displays the status of each tape drive in the libraries connected to DIVA, the tape mounted in each drive, and current operations being performed on a tape in a drive.

If DIVA encounters a problem with a particular drive, it sets the drive to Offline as a safety measure. When a drive is set to Offline, DIVA doesn't use it.

Note: If a drive has been set to Offline, you must address the cause of the error before setting the drive back to Online.

Viewing the DRIVES View

To view the DRIVES view, do the following:

1. Browse to *Resources Management* > *Drives*.

   DIVA opens the *DRIVES* view. The top-right of the *DRIVES* view includes a button for toggling between the table view and the item view. The last view used is the initial view shown on login.

Tapes Tab

When clicked, DIVA opens the *TAPES* view.

Viewing the TAPES View

To view the TAPES view, do the following:

1. Select *Resources Management* > *Tapes*.
2. To display a context menu, click the three vertical dots.
3. To filter search results, click the *Filter* button located at the top left of the view.

4. (Optional) To perform various tape operations, select one or more check boxes for the tapes.

   DIVA displays action buttons at the bottom of the view.
5. To view a menu of options available for the selected tape, click the *More* menu.
6. To display the *TAPE PROPERTIES* view, Click the tape name.

   DIVA displays the *TAPE PROPERTIES* view. This view is for informational purposes only. You can't edit the information.

Tape Group Clones

Users can view the objects on the source tape and clone tapes before actually cloning the tape.

During the first clone of a source tape, an empty tape contained in the set associated with the clone-tape group is selected. The selected clone tape must be of the same type as the source tape, and equal to or greater in size. The selected clone tape is marked as protected.

Only clone jobs can write to a clone tape. DIVA terminates attempts to write to a clone tape by any other job type. The format of the clone tape must be the same as that of the source tape. Every subsequent clone of the same source tape writes to the same clone tape.

Verify that the clone tape contains the same content as the source tape by viewing the clone tape after the clone job completes. The remaining space, number of elements, objects and positions all should match the source tape.

The synchronized state of the source tape is updated to show that the source tape is synchronized with its clone.

If a clone tape holds a number of objects different from the number on the source tape, the source tape isn't synchronized with its clone; and DIVA doesn't display the synchronized flag on the clone tape in the *Tape List* view.

Cloning Missing Content Exclusively

To exclusively copy missing content from a source tape to a destination tape, do the following:

1. Click the source tape containing the newly archived objects.
2. Select *Clone Tape*.

   DIVA displays the *Clone Tape* pop-up form.
3. Click *Submit*.

Cloning Tape Groups Manually

1. (Optional) To display the action button at the bottom of the view, select the checkbox for the tape.
2. To manually clone a tape group, do one of the following:

- Select *Clone Tape* from the three dots next to the desired tape bar code.
- Select *Clone Tape* from the drop-down menu.

  After one or more archives to a tape group containing a clone tape group, you can manually clone the tape containing the newly archived objects. For example, upon archiving of six objects to a tape group called *GroupA* configured with a clone tape group named *GroupB*, you can clone the tape containing the newly archived objects.

Cloning Tape Groups Automatically

To set up the periodic cloning of all tape groups configured for cloning, do the following:

1. Browse to *Configuration* > *General Settings* > *Media*.
2. Enter a value into the field, *Manager: Frequency of Automated Clones (Every X Hours)*.

   This value sets the number of hours between the submission of clone jobs.
3. Enter a value into the field, *Manager: Maximum Simultaneous Automated Clones*.

   This value sets the maximum number of simultaneous clones. DIVA clones tapes in order of least recently cloned. DIVA attempts a clone job only if virtual objects were written to the tape after the last clone.

Repacking Tape Automatically or Manually

When DIVA writes an object to a tape, DIVA writes the object to the place right after the last object written on that tape. When you delete an object from a tape, DIVA can't reuse the space freed by the deleted object. As you delete more and more objects, a considerable amount of unusable space opens in the tape library. This is called tape fragmentation.

You can reclaim this unused space by repacking the tape. The repacking process reads all material from the tape being repacked to a temporary cache; and then writes it back, sequentially, to a new tape in the same tape group as the original.

You can perform a manual tape repack by clicking the *Repack Tape* button. Alternatively, you can trigger repacking automatically, when tape fragmentation and used capacity thresholds are exceeded.

To enable or disable Automatic Repack, do the following:

1. Browse to *Resources Management* > *Tapes*.
2. Check the box(es) next to the desired tapes.
3. To display the form, click the *Automatic Repack* button at the bottom of the tapes list.

Note: Automatic Repack ignores WORM Media. If WORM Media is repacked manually, the space is not recoverable.

---

4. To enable Automatic Repack check the *Automatic Repack Enabled* box, fill in the desired *Tape Filling Threshold*, *Tape Fragmentation Threshold*, *Start Time*, *Duration*, and select the groups from which to repack.
5. Click the *Submit* button.
6. To disable Automatic Repack, deselect the *Automatic Repack Enabled* box.
7. Click the *Automatic Repack* button at the bottom of the tape list.
8. Deselect the check box.
9. Click the *Submit* button.

Troubleshooting Button

When clicked, the *Troubleshooting* button opens the *Troubleshooting* menu, displaying several tabs.

When you click any of the following tabs, DIVA displays the content indicated. This can help in troubleshooting system issues.

Logged Events Tab

When clicked, DIVA opens the *LOGGED EVENTS* view.

This view typically is used with the *JOBS* view for troubleshooting purposes. Using the filters at the top of the view, you can filter the displayed results by *Dates* (Start and End dates and times), *Severity* (Information, Warnings, Errors, and Critical), *Job ID*, and *Description*.

When a particular job fails, you can export the log of that job to a text file and send it to Telestream Support (when requested). Also, the Telestream Support Engineer can collect this information with the Customer Information Collection Tool.

When you run a query for the failed job's Job ID (usually retrieved from the Jobs view), it shows the same events of that job's event log. You can save this file as a text file by selecting *Export*.

DIVA stores a maximum of one million events in its database. When the number of logged events exceeds this value, DIVA begins overwriting the existing events, beginning with the oldest entry.

Drive Alert Logs Tab

When clicked, DIVA opens the *DRIVE ALERT LOGS* view.

This view lists errors reported by tape drives. This information is vendor-specific and may vary depending on the make and model. You can filter your search using the filters at the top of the view. For example, you can look for errors related to a particular tape.

Library Alert Logs Tab

When clicked, DIVA opens the *LIBRARY ALERT LOGS* view.

This view lists errors reported by direct-attached SCSI protocol Managed Storage. This information is vendor-specific and may vary depending on the library make and model. You can filter your search using the filters at the top of the view.

Analytics Charts Tab

When clicked, DIVA opens the *ANALYTICS CHARTS* view.

You can download the data from any of the graphs listed on the *ANALYTICS CHARTS* view into CSV or PNG files.

The *ANALYTICS CHARTS* view displays links to the following graphs:

Operations. Displays a breakdown of your operations over a period of time.

Data Transfers. Displays a breakdown of your data transfers over a period of time, in gigabytes per operation type.

Drive Operations. Displays the number of your drive operations over a period of time. You can select which drives to display in the chart with the filter control.

Tape Status. Displays a breakdown of the current statuses of your tapes: Empty, Full, or Partially Filled.

Resource Utilization. Displays the frequency that resources are used by Actor. You can filter the data in these graphs by actor or group of actors. *Resource Utilization* displays the following charts:

The *Number of Operations* graph displays the number of tape, disk, and storage operations that have been performed over a definable period of time.

The *Time* graph displays the time spent performing tape, disk, and storage operations that have been performed over a definable period of time.

The *Transfer Size* graph displays the amount of data transferred while performing tape, disk, and storage operations that have been performed over a definable period of time.

Storage Trend. Displays the amount of data, in gigabytes, that has either been added or deleted from the system over a definable period of time.

Storage Distribution. Displays the distribution of content across storage types.

Tape Distribution. Displays how the content is distributed across tape groups and the number of in-use tapes for each group.

Content Distribution. Displays breakdowns of where your content is stored, in the following graphs:

The *Instances - Count Distribution* graph displays the number of instances and how they are distributed across disk, tape, and cloud.

The *Instances - Size Distribution* graph displays the cumulative size of the instances and how they are distributed across disk, tape, and cloud.

The *Collection - Disk/Tape/Cloud Count Distribution* graphs display the number of instances per collection and how they are distributed across disk, tape, and cloud with a graph per storage type.

The *Collection - Disk/Tape/Cloud Distribution* graphs display the cumulative size of the instances per collection and how they are distributed across disk, tape, and cloud with a graph per storage type.

See also Analytics in DIVA 9.3.

Migrations Button

When clicked, the *Migrations* button opens the *Migrations* menu, displaying two tabs.

See *Migration Tool*.

The *Migrations* menu includes the following commands:

Start New Migration Task. When clicked, DIVA opens the *SELECT MIGRATION TYPE* view.

Migration Tasks. When clicked, DIVA opens the *MIGRATION TASKS* view.

Configuration Button

When clicked, the *Configuration* button opens the *Configuration* menu, displaying a number of tabs.

User Management Tab

When clicked, DIVA opens the *User Management* menu.

Note: Creating, editing, and deleting DIVA users requires that you log in as a user with a System Administrator (sysadmin) profile.

Managing User Accounts

To manage user accounts, configure your individual user settings, and manage your DIVA license, do the following:

1. Browse to *Configuration* > *User Management* > *User Accounts*.

   DIVA displays the *USER ACCOUNTS* view. This includes a table of DIVA users, including user names, session timeouts, user roles and a column of Actions buttons.
2. Click the Actions buttons desired.

Adding a New DIVA User

To add a new DIVA user, do the following:

1. Browse to *Configuration* > *User Management* > *User Accounts*.

   DIVA displays the *USER MANAGEMENT* view.
2. To add a new DIVA user, click the *Add User* + Add User button below the table.

   DIVA displays the *Add User* dialog.
3. Update the following fields:

User Name (required). Enter the username for this new DIVA user. Permitted characters are letters and numbers, between 5-30 characters—no punctuation marks are permitted. The user name must be unique in the system.

Password & Retype Password (required). Enter the character string in both fields that constitutes the user's secret string to authenticate them in DIVA.

Session Timeout (required; default: 1440). The maximum number of minutes the user can be logged in without using the system before the connection times out. After this, the user must log in again.

Role (required). The role that the user assumes as a DIVA user. Each role has certain privileges and limits.

Click *Save* to create the new user, or *Close* to cancel without saving the record.

See *DIVA User Roles*.

Note: The Save button is gray until all fields meet the criteria. When fields have been entered correctly, the Save button activates and displays blue.

DIVA User Roles

The roles provide varying levels of access and rights. DIVA provides five user roles, identified by these keywords:

- *sysadmin (*system administrator). Users with a sysadmin role can perform all operations in DIVA.
- *admin* (administrator). Users with an admin role can perform all operations except user management.

- *advoperator* (advanced operator). The difference between the operator and advanced operator roles is that the advanced operator can execute Insert and Eject commands.
- *operator*. You use the operator role for normal operations except Insert and Eject commands.
- *user*. Personnel with a user role can perform only user operations.

Note: By default, DIVA includes a sysadmin account with the default password. Change the sysadmin password after installation. If you require additional accounts, you must create them.

For details about configuring role permissions or other user data with the DIVA Web App, see *Configuring Your User Settings*.

License

- Active License
  - Displays the active license information.
- License History
  - Displays historical license information for previously imported licenses.
- Import License
  - Enables importing a new DIVA license.
- Version

Displays information on the current active license including DIVA Version, DIVA Web Version, DIVA REST API Version and Client ID.

License During an Upgrade

During a DIVA upgrade, the DIVA installer configures the license to `manager.conf`. When the Manager service starts, the Manager imports the license.

When upgrading from DIVA 9.1 to DIVA 9.3, the Manager skips the license import to DIVA 9.3 because DIVA 9.1 already has an active license.

Editing a User Configuration

After you have created a DIVA user, only sysadmins can change their settings, including resetting passwords. To edit a user's configuration, do the following:

1. To display the *USER MANAGEMENT* view, browse to *Configuration* > *User Management* > *User Accounts*.
2. Click the *Edit* button on the user record in the table.

   DIVA displays the *Edit User* dialog, containing the user details. The *Edit User* dialog is identical to the *Add User* dialog.

   See *Adding a New DIVA User*.

3. Update the fields.
4. To modify the user record, click *Save*.

Note: If you change user passwords, the next time the users log in, they must supply their new passwords.

Deleting a User

To delete a DIVA user, do the following:

1. Browse to *Configuration* > *User Management* > *User Accounts*.
2. Click the *Delete* button on the user record in the table.

   DIVA displays the *Delete User* dialog.
3. To delete the DIVA user, click *Conform*.

   The DIVA user can continue to use DIVA until logging out.

Configuring Your User Settings

To specify your own user preferences, do the following:

1. Browse to *Configuration* > *User Management* > *User Settings*.
2. (Optional) To reset all of your settings to default values, click the *Reset* Reset button.

The reset can apply to all views using the slider button at the top of the view (Apply To All), or only to specifically selected views using the slider buttons next to each view.

Note: You can adjust tables in DIVA, including determining which columns to display, column order, column width, and total table size. You can make these changes at any time and DIVA remembers them automatically.

LDAP Authentication

You can authenticate DIVArchive with LDAP authentication instead of authenticating against the DIVA database.

See *Launching DIVA Web App and Connecting to DIVA*.

See also *LDAP Authentication* in the *DIVA Installation and Administration Guide*.

Resources Tab

When clicked, DIVA opens the *Resources* menu.

System Settings View

This view and its sub-views allow System Administrators to configure the following:

- Productions Systems (add, edit, or delete)
  - Production systems allow dedication of a particular Actor for specific destinations.
- Sites (add, edit, or delete)
  - Additional sites are optional and may be considered by the Manager for optimal resource.
- Actors (add, edit, clone, delete, or move)
  - Define Actor host definitions and logical functions.
- Unmanaged Storage (add, edit, clone, delete, or move)
  - Define where Core archives from and restores to.
- Transcoders (add, edit, or delete)
  - Define Core transcoders and analyzers.
- Actor-Proxy Connections (add, edit, or delete)
  - Define links between Actors and Proxy Actors.

Library Storage View

Robots

The Robots view displays the robots and libraries--and their properties--in DIVA.

The ROBOTS view displays two tables: ROBOTS and LIBRARIES.

The ACTIONS column displays a More menu to perform operations on robots.

Adding, Editing, or Deleting a Robot

Note: You must be an administrator perform these operations.

To add, edit, or delete a robot, do the following:

1. Click the More menu of the target robot.

   DIVA displays the property view.
2. Edit the name, address, port, and site as desired.

   When you delete a robot, DIVA displays a confirmation dialog.
3. If DIVA displays a confirmation dialog, click *Confirm* to delete the robot, or *Cancel* to cancel the action.

Synchronizing Tape Types (Synchronize DB)

To calculate the block size or total size of the tape types, trigger DIVA to synchronize the tape types so you can enter the values.

To synchronize tape types, do the following:

1. Click the *More* menu of the target robot.

2. Click the *Synchronize DB* Synchronize DB button at the top of the view.

   DIVA displays the *Synchronize DB* dialog. You can specify the robots using the menu.

3. Check the actions you want to execute.

   You can also execute specific actions on the *Tape Types* and *Media Compatibilities* views.

4. To proceed, click Confirm.

   DIVA executes the synchronize operations sequentially and displays their status; either Queued, Running, Skipped, Complete, or Failed.

   If an action fails, hover over the icon to display the error message for the action.

   When synchronizing tape types, if types are found that aren't configured, DIVA displays a dialog where you determine whether to add them to DIVA.

5. To add and configure types, add available types from the left window to the right window.

6. Set the total size and block size for each added type by clicking on the blue Edit button on the top right corner of the right window.

   DIVA displays a dialog.

7. Enter the correct size values for the tape type.

Synchronizing Library Associations

1. To synchronize the robot manager library associations, click the More ⋮ menu of the target robot.
2. Select *Synchronizing Library Associations*.

   DIVA displays a dialog to confirm you want to proceed with the operation.

3. To continue, click *OK*.

   DIVA executes the synchronize operation and displays the status.

Synchronizing Tapes Lists

1. To synchronize the tape lists against the robot manager, click the More ⋮ menu of the target robot.
2. Select *Synchronize Tapes Lists*.

   DIVA displays a dialog to confirm you want to proceed with the operation.

3. To continue, click *OK*.

   DIVA executes the synchronize operation and displays the status.

Synchronizing Drive Types List

To calculate the block size or total size of the tape types, trigger DIVA to synchronize the drive types so you can enter the values.

Again this needs to be done because a user needs to enter the drive type block size.

1. To synchronize the drives list against the robot manager, click the More ⋮ menu of the target robot.

2. Select *Synchronize Drives List*.

   DIVA displays a dialog to confirm you want to proceed with the operation.

3. To continue, click *OK*.

   DIVA executes the synchronize operation and displays the status.

   When synchronizing drive types, if types are found that aren't configured, DIVA displays a dialog where you decide whether to add them to DIVA.

4. To configure types, add available types from the left window to the right window.
5. Click the blue edit button

   DIVA displays a dialog.

6. Enter the block size for each type.

LIBRARIES Table

The *LIBRARIES* table displays each library in DIVA: its name, robot name, library, LSMS, serial number, status and type.

The *ACTIONS* column displays a More ⋮ menu to perform operations on your libraries.

Editing or Deleting a Library

You can edit a library's Name, Serial Number, Description, and Status. After running Synchronize DB (*Synchronizing Tape Types (Synchronize DB)*), edit the library to specify the serial number, and to bring the library online.

When you delete a library, DIVA displays a confirmation dialog. To delete the library, click *Confirm*.

Drives

The *DRIVES* view displays two tables: a *DRIVES* table and an *ACTORS CONNECTED* table, displaying the Actor connections to the currently selected drive in the *DRIVES* table.

The *DRIVES* table displays each drive in DIVA: its name, serial number, library, enabled operations, status, tape, and used status.

The *ACTIONS* column displays a More ⋮ menu to perform operations on your drives.

Editing and Deleting Drives

1. To modify a drive's configuration, click Edit

   DIVA displays a dialog where you can edit the name, serial number, library, LSM, type, block size, and enabled operations.

2. Enable or disable the Used status, as desired.
3. To update this drive's configuration, click *Save*.
4. (Optional) To restore the drive's configuration, click *Reset*.
5. (Optional) To delete a drive, click *Delete* on the target drive.

   DIVA displays a confirmation dialog to confirm that you want to delete the drive from DIVA.

6. To confirm the operation, click *OK*.

Adding a Drive-Actor Connection

1. To add an Actor connection to the target drive, click the *Add Actor Connection* button.

   DIVA displays a form, enabling you to select the Actors you want to connect to the drive. You can select or connect some or all Actors.

Connected Actors

The *CONNECTED ACTORS* view displays all Actor Connections to all Drives in the system. To see the connections available, you can filter the view by either Actor or Drive.

Media Compatibilities

The MEDIA COMPATIBILITIES view displays the media compatibilities that have been detected and configured by Synchronize DB. It is also possible to synchronize just the media compatibilities again if the list is incomplete.

Editing or Deleting Media Compatibilities

To either edit or delete the compatibility, do the following:

1. Click the three dots next to the desired media.

   When editing a media compatibility you can edit whether it is Read/Write or Read Only.

   DIVA displays a confirmation dialog before deleting a media compatibility.

2. To delete the compatibility, click *Confirm*, or, to cancel the action, click *Cancel*.

Tape Types

The *Tape Types* panel displays the tape media types in the system that were detected when performing a Synchronize DB. If the list appears to be incomplete, perform another synchronization of just these types—click the Synchronize Tape Types button.

Tape Type Size and Deletion

When editing a tape type, you can edit the total size of the type. The value is displayed in the most appropriate display size, but this can be entered in any size unit when editing.

DIVA displays a tape type confirmation dialog before deleting. Click Confirm to delete the compatibility, or Cancel to cancel the action.

Tape Groups

The *Tape Groups* panel displays the tape groups currently defined in the system. Actions available are Add, Edit, Clone, and Delete.

Tape Group Edits or Deletions

When adding or editing a tape group, users can set the Name, Set Id, Description, Tape Format, Clone Group, Number of Tapes Used For Repack, and whether Encryption, Compression, Worst Fit Writing, or Verify Write are enabled for the group.

DIVA displays a confirmation dialog before deleting a tape group,. To delete the compatibility, click *Confirm*, or, to cancel the action, click *Cancel*.

Empty Ejected Tapes

Empties ejected tapes.

Unused Tapes (new topic)

Lists unused tapes.

Displaying Library Storage

Library storage panels enable users with System Administrator and Administrator privileges to manage and configure the DIVA library storage components.

To display the Library Storage panels, do the following:

1. Browse to *Configuration* > *Resources* > *Library Storage*.

   DIVA displays the DIVA library storage components.

Note: Operators, Advanced Operators, and Users do not have access to these library storage component operations.

Storage Plans Settings View

When clicked, the *Storage Plans* button opens the *STORAGE PLANS SETTINGS* view.

Cloud Storage View

This view and its sub-views allow system administrators to configure the following:

- Cloud Accounts (add, edit, clone, or delete)
  - Define storage accounts to connect to cloud storage.
- Cloud Buckets (add, edit, clone, or delete)
  - Define Cloud Buckets that should be accessible in DIVA.
  - Connected Actors (add, edit, clone, or delete)
- Connected Actors (viewing only)
  - View all Actor connections to Cloud Buckets.

Setting the Cloud Quota

DIVA doesn't display Cloud storage in the system dashboard unless you set a quota limit on the Cloud Bucket. The quota limit is for display purposes on the dashboard

only. If you don't set the quota limit, then Cloud Disk displays only the amount of space used.

To set the Cloud Quota, do the following:

1. Browse to *Configuration* > *Cloud Storage* > *Cloud Buckets* > *Add Cloud Bucket* (or *Edit Cloud Buckets*).
2. Adjust the setting as desired.

Disk Storage View

This view and its sub-views allow System Administrators to configure the following:

- Arrays (add, edit, clone, or delete)
  - Define disk arrays that should be accessible in DIVA.
- Disks in DIVAgrid (add, edit, clone, or delete)
  - Define Disks that belong to the disk arrays accessible in DIVA.
- Connected Actors (viewing only)
  - View all Actor connections to disk arrays.

Proxy Storage View

When clicked, the *Proxy Storage* button opens the *PROXY STORAGE* view. The *PROXY STORAGE* view displays the following buttons:

Arrays. Define disk arrays for proxy storage.

Filters. Define Filters to control which objects have proxies generated

Configuration. Configure and control the proxy service

Proxy Jobs. View and reprocess proxy creation jobs

Attachment Storage View

When clicked, the *Proxy Storage* button opens the *ATTACHMENT STORAGE* view. The *ATTACHMENT STORAGE* view displays the following buttons:

Arrays. Define Disk Arrays for attachment storage.

Attachment Archiving. Configure archiving of attachments from metadata store to attachment array storage.

Attachment Status. Manage the archiving and extraction processes for attachments.

General Settings View

When clicked, the *General Settings* button opens the *GENERAL SETTINGS* view.

Use this view to manage and configure the these components (System Administrators only):

- **Checksums**—Configure creation of checksums by manager type.
- **Media**—Configure automated clones of media.
- **Objects**—Configure how the manager handles complex objects.
- **SMTP Notifications**—Configure email notifications.

  See *Configuring SMTP Messages* in the *DIVA Installation and Administration Guide*.
- **Security**—Configure the use of encryption keys.

If you have made any changes, click Save All Changes to update DIVA.

Services Tab

When clicked, the *Services* tab opens the *Services* menu, displaying several tabs.

Analytics Tab

When clicked the *Analytics* tab opens the *ANALYTICS* view.

Database Backup Tab

When clicked the *Database Backup* tab opens the *DATABASE BACKUP* view.

Metadata Tab

When clicked the *Metadata* tab opens the *METADATA* view.

API Translation Tab

When clicked the *API Translation* tab opens the *API TRANSLATION* view.

Workflow Tab

When clicked the *Workflow* tab opens the *WORKFLOW* > *CONFIGURATION* view.

Log Out Button

When clicked, the Log Out button logs you out of the DIVA Web App and out of DIVA.

Migrations Menu

Clicking on any of the following submenus will display the content indicated. See Migration chapter for more details.

Start New Migration Task

This page allows users to begin a copy or move migration job. Users follow through the wizard pages to complete the task. The wizard automatically moves to the next page when the user makes a selection on each page and continues to completion.

Migration Tasks

To view the status of the current migration task and previously executed migrations. do the following:

1. Browse to *Migrations* > *Migration Tasks*.

   DIVA opens the *MIGRATION TASKS* view.

2. To view the properties of a migration task, click its name.

   DIVA opens the *MIGRATION TASK PROPERTIES* view.

3. To refresh the view and update the status, click the Refresh button towards the top-right of the view.

Local Settings

This page and its sub-pages allow system administrators to configure the following:

- Date and Time
  - Customize how dates and times are displayed.
- Customization
  - Customize general settings for the website.
- Reset Local Config

Reset custom configuration for individual pages.

Jobs

This chapter describes various types of DIVA jobs.

Topics

- Setting the Job Files Path Root and Files Parameters for an Archive
- AXF Archive and Restore Jobs
- Partial File Restore Jobs
- Delete and Delete Instance Jobs
- Stage Jobs
- Copy Jobs
- Copy As Jobs
- Quality of Service

Setting the Job Files Path Root and Files Parameters for an Archive

The *Files Path Root* and *Files* parameters in the *Archive Submission* form work together. Identify a logical business object before filling in these parameters and executing the job.

To set the Files Path Root and Files parameters for an archive, do the following:

1. Enter the path to the main file folder (the top folder) into the *Files Path Root* field. For example, `C:\DROPFOLDER\Media\Virtual Object1\`.
2. Enter the individual files and any additional sub-folders into the *Files* text box; for example, `subfolder1\file3`.

   The *Files* field can contain an absolute path. Telestream doesn't recommend this, however, because it prevents the object from being restored to a different root folder.

   Assuming a *Files Path Root* is identified, don't use the full file path in the *Files* field. Use only the folder names and file names that are located under the identified *Files Path Root* folder.

Note: Alternatively, leave the *Files Path Root* field blank, and enter the full file path and name into the *Files* field.

The following examples show how you can use these parameters.

Correct Examples

The following entries archive only the specified files located in `C:\DROPFOLDER\Media\Object1\ and the subfolder1\file3`.

- `Files Path Root`

  `C:\DROPFOLDER\Media\Object1\`

- `Files`

  ```
  file1
  file2
  subfolder1\file3
  ```

The following entries archive all folders and files located in `C:\DROPFOLDER\Media\Object1\`.

- `Files Path Root`

  `C:\DROPFOLDER\Media\Object1\`

- `Files`

  `*`

The following entries are correct but not recommended, because in the future the object can't be restored to a different location. The system loses flexibility and compatibility with other storage devices and in some scenarios, transcoding and Partial File Restore capabilities are also limited. In this example, the Files Path Root was left blank and the absolute paths are entered in the Files field.

- `Files Path Root`
- `Files`

  ```
  C:\DROPFOLDER\Media\Object1\file1
  C:\DROPFOLDER\Media\Object1\file2
  C:\DROPFOLDER\Media\Object1\subfolder1\file3
  ```

Incorrect Example

The following entries result in an error and the Archive job doesn't complete:

- `Files Path Root`

  `C:\DROPFOLDER\Media\Object1\`

- `Files`

  ```
  C:\DROPFOLDER\Media\Object1\file1
  C:\DROPFOLDER\Media\Object1\file2
  C:\DROPFOLDER\Media\Object1\subfolder1\file3
  ```

Archive Job with Delete on Source

There are instances where you must delete content, and possibly the parent folder, on a server. There are two options available to satisfy all possible scenarios:

- `-r`

  Recursive delete
- `-delete_fpr`

  Recursive deletion including the parent folder

The two options work either separately or together as indicated in the following workflow examples:

Example 1

DIVA deletes the content of C:\source\root recursively because of these settings:

- `Files Path Root`

  `C:\source\root`
- `Files`

  `*`
- `Options`

  `-r`

Example 2

DIVA deletes the content of C:\source\root recursively and the parent folder (root) because of these settings.

- `Files Path Root`

  `C:\source\root`
- `Files`

  `*`
- `Options`

  `-r -delete_fpr`

Example 3

DIVA deletes only the content of C:\source\root because of these settings.

- `Files Path Root`

  `C:\source\root`
- `Files`

  `*`
- `Options`

Example 4

DIVA deletes only the content of C:\source\root, and eventually the parent folder (root) if it is empty, because of these settings.

- `Files Path Root`
  `C:\source\root`
- `Files`
  `*`
- `Options`
  `-delete_fpr`

Example 5

DIVA deletes the content of C:\source\root\Object recursively and the parent folder (Object) because of these settings.

- `Files Path Root`
  `C:\source\root`
- `Files`
  `Object\*`
- `Options`
  `-r`

Example 6

DIVA deletes the content of C:\source\root\Object recursively, then delete C:\source\root\Object, and finally delete C:\source\root if it is empty because of these settings.

- `Files Path Root`
  `C:\source\root`
- `Files`
  `Object\*`
- `Options`
  `-r -delete_fpr`

Example 7

DIVA deletes the content of C:\source\root\Object1 recursively, delete C:\source\root\Object1, delete the content of C:\source\root\Object2 recursively, and delete C:\source\root\Object2 because of these settings.

- `Files Path Root`
  `C:\source\root`
- `Files`
  `Object1\*`
  `Object2\*`
- `Options`
  `-r`

Example 8

DIVA deletes 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, and delete C:\source\root if it is empty because of these settings.

- `Files Path Root`

  `C:\source\root`

- `Files`

  ```
  Object1\*
  Object2\*
  ```

- `Options`

  `-r -delete_fpr`

Example 9

DIVA deletes the content of `C:\source\root\Object1` recursively, delete `C:\source\root\Object1, delete 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, and delete `C:\source\root` if it is empty because of these settings.

- `Files Path Root`

  `C:\source\root`

- `Files`

  ```
  Object1\*
  Object2\*
  ```

- `Options`

  `-r -delete_fpr`

Launching a Restore Job

A restore job is the transfer of an object to a target server. To launch a restore job from the DIVA Web App, do the following:

1. Browse to the *Content Management* > *Catalog Browsing* view.
2. Click the three dots next to the object to restore.
3. Select the restore type from the context menu.
4. Fill out all required information on the form.

   The following fields are included in the *Restore Job* form:

- **Object Name**

  The name of the object to be restored.

- **Object Collection (Previously *Category*)**

  The collection assigned to the object when it was archived. This parameter can be left empty but this may result in an error if several objects have the same name.

- **Instance**

  If multiple instances of an object reside in DIVA, the particular instance to restore can be specified. If left blank, DIVA selects the instance that provides the most optimum transfer.

- **Destination Servers (required)**

  Destination (for example, a video server or browsing server) for the object files. This name must be known by the DIVA configuration. Use the drop-down list to select the desired Destination.

- **Quality Of Service (required)**

  For detailed descriptions, see *Quality of Service*.

- **Additional Services**

  Additional and optional services are available using the Additional Services pull-down list (to the right of the Quality of Services). To request those services select from the following options:

  – Default

    Operate using the default setting in the DIVA configuration.

  – Do Not Overwrite

    Do not overwrite existing files on the destination server.

  – Do Not Check Existence

    Do not check for the existence of the clip on the server.

  – Delete And Write

    Force delete and rewrite if object exists on the server.

- **Options**

  Additional options for performing the transfer of data from DIVA to the target server. These options supersede any options specified in the DIVA configuration database. The possible values for Options follow:

  – No Entry

    No entry in this field specifies no options.

  – `-login`

    A user name and password is required to log in to some sources. This option obsoletes the -gateway option from earlier versions.

  – `-pass`

    The password used with -login.

- **Files Path Root**

  The root folder on the destination where the object files are placed. This option appends or overrides the FPR used in the original archive job. If left empty, the files are placed in the Files Path Root folder specified when archiving the object.

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the Priority Value text box. If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

  There are six predefined values as follows:

  - `Min`
  - `Low`
  - `Normal`
  - `High`
  - `Max`
  - `Default`

  If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

5. To execute the job, click *Submit*.

This process submits an object Restore job to DIVA and the DIVA chooses the appropriate instance to be restored. The job fails if the requested object is on media that is not available.

Multiple Restore Jobs

You can restore an object to multiple destinations simultaneously with a multiple restore job. With a multiple restore job, you specify all the necessary destinations in one command in a single job, instead of specifying a separate standard restore job for each destination. This is beneficial when restoring from a tape instance, because the tape is read once for the multiple restore job rather than multiple times for many single restore jobs of the same object. DIVA supports up to five simultaneous destinations.

If the object to be restored is part of a spanned tape set, restore it to cache before transferring it to all destinations. If the transfer to one of the destinations fails, the others still proceed (if possible) and DIVA displays the job status *Partially Aborted*.

If multiple renaming rules are defined, DIVA processes the rule for each server independently.

Restoring to Multiple Destinations

A destination here refers to the unmanaged storage that is selected in the multiple-restore job.

To execute a Multiple Restore job, do the following:

1. Browse to *Content Management* > *Catalog Browsing*.
2. Click the three dots next to the object to be restored.

   DIVA displays a form where you can set required parameters.
3. Enter, or select, all required parameters on the form.
4. Use the *Destination Servers* drop-down menu to select the first destination server.
5. Click the plus icon (green +) next to the *Files Path Root* text box.

   This adds the selected server to the list in the text area below *Files Path Root*.
6. Continue adding servers in the manner described above until all servers are selected.
7. To execute the job, click *Submit*.

Copying an Object

To copy an object, do the following:

1. Navigate to the *Content Management* > *Catalog Browsing* view.
2. Click the three dots next to the object to be copied.
3. Select Copy from the menu.
4. Fill in the fields in the form.
5. Click Submit to execute the job.

The following fields are included in the Copy job form:

- **Object Name**

  The name of the source object. This is automatically populated with the Object Name selected.

- **Object Collection (Previously *Category*)**

  The Collection of the source object.

- **Destination Media (required)**

  The Destination Media can be either a tape group or disk array. Use the pull-down list to select the Destination Media to use. If the new instance is being created in the same tape group or array, the job only succeeds if it can be copied to a separate tape or disk.

- **Instance**

  If multiple instances of the object reside in DIVA, use the menu to select which specific instance to copy. If no instance is specified, DIVA selects the instance that provides the most optimal execution time.

- **Options**

  Additional options for copying the object in DIVA. These options supersede any options specified in the DIVA configuration database.

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the Priority Value text box.

  There are six predefined values as follows:

  - `Min`
  - `Low`
  - `Normal`
  - `High`
  - `Max`
  - `Default`

  If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

Copying an Object as a New Object

To copy an object as another object, do the following:

1. Navigate to the *Content Management* > *Catalog Browsing* view.
2. Click the three dots next to the object to be copied.
3. Select *Copy As* from the menu.
4. Fill in the fields in the form.
5. Click *Submit* to execute the job.

For any object, the Object Name may not necessarily match that of the file name of the essence stored within it. If you use the Copy As command to create an object, it still is restored using the same name with which it was originally archived.

For example:

If a file named *xyz* is archived from a server, regardless of what Object Name was given to it in DIVA, it always restores to a destination as *xyz* regardless of its Object Name.

When DIVA issues a Copy As job, the *Instance* field of the job is left empty. However, you can select an instance number from the drop-down list before DIVA issues the job. If you leave this field empty, and there are multiple instances of that object, DIVA selects the instance that completes the transfer in the shortest possible time by default. For example, DIVA selects a disk instance over a tape instance. This depends on the QOS specified and the DIVA configuration.

The following fields are included in the *Copy As Job* screen:

- **Object Name**

  The name of the source object. This is automatically populated with the Object Name selected.

- **Object Collection (Previously *Category*)**

  The Collection of the source object. This is automatically populated with the Object Collection selected.

- **Destination Object Name (required)**

  The name of the destination Virtual Object.

- **Destination Object Collection (required)**

  The Collection of the destination object.

- **Destination Media**

  The Destination Media can be either a tape group or disk array. If the new instance is being created in the same tape group or array, the job only succeeds if it can be copied to a separate tape or disk. Use the pull-down list to select the Destination Media.

- **Destination Storage Plan**

  The storage plan to assign to the new object on the destination. Use the pull-down list to select the appropriate storage plan.

- **Instance**

  If multiple instances of the object reside in DIVA, use the pull-down list to select which specific instance to copy. If no instance is specified, DIVA selects the instance that provides the most optimal execution time (for example, a disk instance is selected over a tape instance).

- **Comments**

  Comments added here are added to the new object's properties.

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the *Priority Value* text box.

  There are six predefined values as follows:

  - `Min`
  - `Low`
  - `Normal`
  - `High`
  - `Max`
  - `Default`

  If the *Default Priority* check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

AXF Archive and Restore Jobs

Instead of archiving AXF files, DIVA archives the contents of AXF files, and retrieves the checksums and provenance of the object.

The optional restore parameter, `-axf`, instructs DIVA to restore the original media file into an AXF file. DIVA restores the content into a new AXF Wrapper. When it's combined with `-rm` or `-rxml`, you can use this option to export an object with metadata information,. You can drop this object into a DFM Watch Folder.

The AXF archive and restore functionality includes the following:

- Archives the content of an AXF file using auto-detection.
  - Identifies the *axf* file name extension
  - Confirms it is a single file
  - Checks the beginning of the file for specific AXF properties
  - Checks for metadata information
- Restores an object into a new single AXF file.
- Preservation of checksums
- Preservation of metadata
- Preservation of provenances
- Complex Object support

This options works with FTP_STANDARD, LOCAL, DISK, CIFS, and EXPEDAT servers.

Failure Conditions

DIVA terminates a stage job under the following conditions:

- Object doesn't exist

- Target media doesn't exist
- Availability is beyond range.

DIVA configuration setting for regulating the upper limit for this value as follows:

- `#RELOADABLE in SERVICE mode`

  The number of days an instance is available after it is staged. The default setting of 3 implies a value between 1 day and 3 days is accepted.

- `#MAX_DAYS_FOR_STAGING=3`
  - Priority is in range (1-100)
  - Restricted character is not used
  - Object is available

Partial File Restore Jobs

DIVA supports four types of Partial File Restore. The type implemented is determined by the job's format parameter. This job submits a Partial Object Restore job and DIVA chooses the appropriate instance to be restored. The job fails if the requested object is on media that is not available.

The following list describes each type of Partial File Restore:

- **Files and Folders**

  This type of Partial File Restore enables extracting entire files from the archive or extracting entire directories and their contents. Multiple files and directories can be extracted in the same job. The files are restored with the file names and path names that were specified in the archive. There is no valid renaming option in File and Folder Partial File Restore. For example, a file archived as misc/12-2012/movie.avi is partially restored to a misc/12-2012 subdirectory with the name movie.avi.

  When a folder is specified in a File and Folder Partial File Restore, all files within that folder (and the folder itself) are restored. Also, each folder can include the `-r` option to recursively restore all folders nested within the target folder.

- **Byte Offset**

  This type extracts a range of bytes from a particular file in the archive. For example, you can extract bytes 1 to 2000 (the first 2000 bytes of the file), or byte 5000 to the end of the file (or both) and restore them to an output file such as movie.avi.

Note: The result of the Byte Offset Partial File Restore is usually not playable when applied to video files. The Actor doesn't apply the header or footer, according to the video format.

- **Timecode**

  This type of Partial File Restore enables selecting a portion of a particular media file based on a timecode. For example, a user could extract from 00:00:04:00 to 00:10:04:00 (a 10 minute segment starting 4 seconds in and ending at 10 minutes

and 4 seconds), and place that segment into an output file such as movie.avi. The resulting file is a smaller version of the original movie file.

Note: The result of the Timecode Partial File Restore is a valid clip when applied to video files. The Actor applies the header and footer, according to the video format. If the Actor can't parse the format, the job is terminated. This type of Partial File Restore can only be applied to a valid video clip.

- **DPX**

  This type of Partial File Restore enables extracting a range of DPX files from the archive. The entire object is viewed as a single media item, with one DPX file representing one frame of media. Only files with a DPX, TIF, or TIFF extension in the archive are considered frames for the purposes of this command.

  The first DPX, TIF, or TIFF file in the archived object is considered frame 1, the second file in the archive is frame 2, and so on.

  In the unlikely event that these files are mixed, the first sequential file of any of the three extensions determines which files are considered to be part of the sequence. For example, if a stray TIF file is mixed with a collection of DPX files and it came first in the sequence, the sequence is interpreted as a TIF sequence and DPX files are ignored, even if this was not the intention.

  For example, to extract frames 10 through 15 using DPX Partial File Restore, it restores the tenth DPX file that appears in the archive, the eleventh DPX file, and so

on, ending with the fifteenth DPX file, for a total of six files. Any other files (such as WAV files) are skipped by DPX Partial File Restore.

Special frame numbers 0 and -1 may be used to refer to the first and last frame respectively. Frame 0 is valid as the start of a frame range and Frame -1 is valid as the end of a range.

Valid frames and ranges are as follows:

- Frame 0 = first frame (select the Start of File check box)
- Frame 1 = the first frame in the sequence
- Frame n = the nth frame in the sequence
- Frame -1 = last frame (select the End of File check box)
- Specifying Frame 0 as the last frame is considered invalid.
- Specifying Frame 0 to 0 is invalid; won't return the first frame as may be intended.
- Specifying Frame 0 to 1 or Frame 1 to 1returns the first frame.
- Specifying the Frame -1 in the first frame produces an error. You also can't specify Frame -1 to -1 to return the exact last frame if the exact number of the last frame is unknown.

Examples:

- `startRange=0 - endRange=1`

  Restores only the first frame.

- `startRange=600 - endRange=635, startRange=679 - endRange=779`

  Restores frames 600 through 635, and frames 679 through 779.

- `startRange=810 - endRange=-1`

  Restores all frames from frame 810 to the end of the archive.

  The actual file name may (or may not) match the frame number in DIVA. After restore DIVA interrogates the archive, finds the file order, and determines the Frame Number from the resulting file order found, it doesn't consider the file-name. The first DPX, TIF, or tiff file found is considered Frame 1.

  Be careful when archiving DPX files to ensure they can be partially restored prop-erly—DPX Partial File Restore doesn't examine the filename or the DPX header information to determine which file is assigned to which frame. The assignment is based on the order in which the DPX files appear in the archive. By default this order is based on ordering established by the source and is typically alphanu-meric. For example, NTFS DISK servers order files and folders case insensitively as a general rule, except where diacritical marks, such as ', `, ^, and so on are applied.

  By default, when DIVA encounters a sub-folder it recursively processes all of the children of that folder (including sub-folders) before continuing with other files. If a folder appears in the alphanumeric folder listing it is archived recursively in the order it appears, but this can potentially create some issues. For example, if you 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.

DPX Partial File Restore uses an entire object as a single piece of media. If multiple reels or clips are in an archive, they can be stored in folders and partially restored using File and Folder Partial File Restore, but to DPX Partial File Restore they are viewed as a single clip. If this is a desired effect, ensure that the directories are sorted alphanumerically in the order the frames should be arranged.

DIVA doesn't perform any special audio handling for DPX media (other than what might be embedded in DPX files). DIVA can support transcoding of DPX media, but a transcoder may change the file names and/or file order of the DPX archive.

Submitting a Partial File Restore Job

To submit a Partial File Restore job in the DIVA Web App, do the following:

1. Browse to *Content Management* > *Catalog Browsing*.
2. Click the three dots next to the object.
3. On the context menu, select *Partial Restore*.

   DIVA displays the *Partial Restore* form with the selected Object Name and Object Collection already filled in.
4. Select the desired type of Partial File Restore from the menu.

   Each type of Partial File Restore has different options, except for Files and Folders Partial File Restore, which doesn't have any options.
5. Double-click the files in the left pane to copy them to the right pane, and to add them to the job.
6. Click *Next*.
7. Enter required parameters for these Partial File Restore types.
8. Click the blue edit icon next to the object name.

   DIVA displays a form for entering the required additional parameters:

   - `Byte Offset`

     No offset is entered until you open the Options dialog box and manually enter them. Add the required offset parameters and click Add to include them in the job; then click OK to return to the job wizard.
   - `Timecode`

     The File Format (required) list is enabled after selecting the Timecode Partial File Restore. Select the proper file format from the list, or leave it set to Autodetect.

     Click the edit icon (blue) next to the object name after it was moved to the right pane. A dialog displays; enter the required additional parameters. Add the required Offset parameters and click Add to include them in the job; then click *OK* to return to the job wizard.
   - `DPX`

     Click the edit icon (blue) next to the object name after it was moved to the right pane. A dialog display, so that you can enter the required additional parameters.

Add the required parameters and click Add to include them in the job; then click *OK* to return to the job wizard.

Note: Partial File Restore jobs for AVI format files must include the same offset range (TCin, TCout) for all object components (for example, clip.avi, clip_1.wav, clip_2.wav).

The following list describes the parameters in the final Partial File Restore job screen:

- **Instance**

  If there are multiple instances of an object, DIVA selects the instance which allows the job to complete in the least amount of time (for example, a disk instance is selected over a tape instance). Specifying an instance number in this field overrides this behavior and target the specific identified instance.

- **Destination**

  Destination (for example, a video server or browsing server) for the object files. This name must be known by the DIVA configuration. Use the drop-down list to select the desired Destination.

- **Files Path Root**

  The root folder on the destination where the object files are placed. If left empty, the files are placed in the Files Path Root folder specified when archiving the object.

- **Options**

  Additional options for performing the transfer of data from DIVA to the target server. These options supersede any options specified in the DIVA configuration database. The possible values for Options follow:

  – No Entry

    No entry in this field specifies no options.

  – `-login`

    A user name and password is required to log in to some source servers. This option obsoletes the -gateway option from earlier versions.

  – `-pass`

    The password used with -login.

- **Quality of Service**

  One of the following codes:

  – `Default`

    Restoring is performed according to the default Quality of Service (direct for restore operations).

  – `Cache Only`

    Use cache restore only.

  – `Direct Only`

    Use direct restore only; no disk instance is created.

  – `Cache and Direct`

    Use cache restore if available, or direct restore if cache restore is not available.

  – `Direct and Cache`

    Use direct restore if available, or cache restore if direct restore is not available.

  For detailed descriptions, see *Quality of Service*.

- **Additional Services**

  Additional and optional services are available. To request those services use the menu to select the following options:

  – `Default`

    Use the default settings.

  – `Do Not Overwrite`

    Do not overwrite existing files on the target server.

  – `Do Not Check Existence`

    Do not check if the object exists on the target server.

  – `Delete and Write`

    Force delete of existing instances on the target server and write a new instance.

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the Priority Value text box.

  There are six predefined values as follows:

  – `Min`

  – `Low`

  – `Normal`

  – `High`

  – `Max`

  – `Default`

  If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

7. Select the Partial File Restore type and associated options for each object.
8. To execute the job, click the *Submit* button.

Delete and Delete Instance Jobs

Use the Delete command to delete all instances of an object, or only a specific instance of the object from DIVA.

Caution: You must use this command with caution; data loss could result.

This command submits an Object Delete job to DIVA and deletes every instance of the object (unless otherwise specified).

The Instance field of the Delete job determines exactly what is deleted from DIVA. If this field is left empty, then all instances of that object is deleted. A specific number entered into this field only deletes the specified instance.

Deleting an Object

To delete an object, do the following:

1. Browse to *Content Management* > *Catalog Browsing*.
2. Click the three dots next to the object name on the view.
3. Select *Delete Object* from the menu.

   DIVA displays a form.
4. Fill in the fields.
5. Check the box indicating you understand that the instance is completely removed from the system.
6. To process the job, click *Submit*.

When the job is done, DIVA updates the Num Instances column.

Note: Deletes and repacks do not clear WORM media because these are Write-Once media. The instances are deleted from the DIVA database, but the space is not recoverable.

The following fields are included in the Send Delete job screen:

- **Object Name**

  The name of the Object to be deleted. This is automatically populated with the Object Name selected.
- **Object Collection (Previously *Category*)**

  The Collection assigned to the object when it was archived. This parameter can be a null string, but this may result in an error if several objects have the same name.

- **Instance**

  If multiple instances of the Virtual Object reside in DIVA, use the pull-down list to specify which specific instance to delete. If no entry is selected in this field, DIVA deletes all instances of that object.

- **Media**

  The media can be an existing tape group or disk array. The menu only displays those items already configured in the DIVA configuration.

- **Options**

  Additional options for deleting the object from DIVA. These options supersede any options specified in the DIVA configuration database.

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the *Priority Value* text box.

  There are six predefined values as follows:

  - `Min`
  - `Low`
  - `Normal`
  - `High`
  - `Max-`
  - `Default`

  If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

Note: Instances on tapes linked to a clone can't be deleted. Attempts to delete an instance or object with instances on a Source or Clone Tape result in the job terminating. The clone Storage Link must be removed to delete the instance.

Stage Jobs

A new endpoint supports the Stage job in the DIVA REST API. The POST endpoint accepts the name of the Object, Collection or Collection Name (previously *Category*), target media to stage the content, target storage class (if not specified it uses the default), restore tier, priority, options and number of days to stage the content. Here is a sample body:

```
{
"ObjectName": "testVirtual ObjectName",
"collectionName": "testCollectionName",
"targetMedia": "testMedia",
```

```
"numDaysAvailable": 3,
"options": "",
"priority": 50,
"restoreTier": "EXPEDITED",
"storageClass": "STANDARD"
}
```

Copy Jobs

Use the Copy command to create an instance of an existing object in the same or another tape group or array. This is useful for creating a backup copy of the object on another medium.

Submits a job for copying an archived object to a new object, with another name and/or Collection, to the Manager and the Manager chooses the appropriate instance as the source of the copy. All types of transfers (disk to disk, disk to tape, tape to disk, and tape to tape) are supported.

In the event the requested object is on media that is not available, the job fails.

When a copy job is issued with no instance specified and there are multiple instances of that object, DIVA selects the instance that executes the copy operation in the shortest possible time (for example, a disk instance is selected over a tape instance). If an instance number is selected in the Instance field of the form, the copy operation uses that specific instance only.

Copy As Jobs

When an object is archived to DIVA, it is uniquely identified by its Object Name and Object Collection. Neither the Name nor the Collection can be altered once it exists within the Core database. The Copy As command allows creation of a new object in DIVA with a new Object Name and/or Collection, and then the original object can be deleted if desired or necessary (the deletion must be performed manually).

Quality of Service

The Quality of Service (QOS) parameter defines how a file is transferred to and from a tape, from a source, or to a destination.

Use QOS with one of the following options:

- Default

  Restoring is performed according to the default Quality of Service (direct and cache for restore operations).

- Cache Only

  Use cache restore only. Cache only restores can only restore from a tape instance. However, a tape instance on a tape in a tape group with a higher priority is preferred to a tape instance on a tape in a tape group with a lower priority.

- Direct Only

  Use direct restore only—no disk instance is created.

- Cache and Direct

  Use cache restore if available, or direct restore if cache restore is not available.

- Direct and Cache

  Use direct restore if available, or cache restore if direct restore is not available.

- Nearline Only

  Use Nearline restore only. Nearline restore restores from a disk instance if one exists. Otherwise, it creates a disk instance and restore from the newly created disk instance. However, a disk instance on a disk in an array with a higher priority is preferred to a disk instance on a disk in an array with a lower priority.

- Nearline and Direct

  Use Nearline restore if available, or direct restore if Nearline restore is not available. However, a disk instance on a disk in an array with a higher priority is preferred to a disk instance on a disk in an array with a lower priority.

Job (Formerly *Request*) Options

- `-qos_direct_only`
- `-qos_cache_only`
- `-qos_direct_and_cache`
- `-qos_cache_and_direct`
- `-qos_nearline_only`
- `-qos_nearline_and_direct`

The DIVA REST API allows a user to configure these settings via the *POST/requests* archive and *POST/requests/restore* endpoints. Refer to the Swagger page for details.

Server (Source/Destination) Connect Options

In addition to job options, you can specify the Quality of Service at the server level, by specifying qos=<qos setting>. Ex: qos=nearline_only.

See also Quality of Service (`qos=`).

The DIVA REST API allows a user to configure these settings via the *POST* and *PUT/*server endpoints. Refer to the Swagger page for details.

Media (Array/Group) Archive and Restore Settings

You can specify the Quality of Service (QOS) at a Media level, by individually specifying the settings.

The DIVA REST API allows a user to configure these settings via the *POST* and *PUT/* groups and */arrays* endpoints. The attributes of the job body are documented in the swagger documentation. Refer to the Swagger page for details.

As the destination media are specified in an Archive submission, the QOS is known at submission of the request and is applied according to the priority of application defined under *Priority of Application of QOS*. However, for Restore object jobs, the Manager determines the instance to use partially based on the QOS. As the QOS is embedded in the source media, the instance is selected using the following strategy.

The usual algorithm is applied using the QOS of the source media of one of the instances of the object. As usual, all instances are considered with that QOS value. If the selected instance has a media with a QOS value that matches the one input into the algorithm, then that instance is selected. If not, then the next QOS assigned to an instance of the source media is used and the process is repeated. If no instance can be found using the above strategy, then both the media QOS and instance are fixed to one of the instances of the object when using the usual algorithm. If still no instance can be found to satisfy the request, then as usual, a delayed or rejected solution is returned.

For a restore instance, the approach is simpler as the instance is known on submission of the job. Therefore, the source media QOS is known in advance and is applied according to the section entitled, *Priority of Application of QOS*.

Priority of Application of Quality of Service (QOS)

The Quality Of Service (QOS) is applied to an archive or restore job using the following prioritization:

- The QOS specified via a job (request) option
- Normal specification via a job (request)
- The QOS specified in the server (source/destination) connect option
- The separate Archive and Restore QOS of the media (group or array)
- The default of DIRECT_AND_CACHE for Archive submissions OR the global setting (DIVAMANAGER_RESTORE_QOS) in the manager configuration file for Restore jobs.

Order of Priority

- Job options take precedence over the normal Quality of Service (QOS) specification.
- The normal QOS specification takes precedence over the Server Connect Options.
- The Server Connect Options take precedence over the Media settings for QOS.
- The Media settings for QOS take precedence over the default values of DIRECT AND CACHE for Archive submissions and the value of DIVAMANAGER_RESTORE_QOS in the manager configuration file for Restore jobs.
- If DEFAULT is specified at any level, then DIVA uses the concrete value of the setting at a lower level. For example, if all specifications above the Media specification of QOS are DEFAULT, but the Media has a QOS value of DIRECT, then DIRECT is used.

Value (API Code)	Job Option	Job Type	Description
default (0)	N/A	BOTH (ARCHIVE AND RESTORE)	Use the QOS specified at a lower level. Ex: if no specification at Job option level, use the specification at server level, …
cache (1)	-qos_cache_only	BOTH	The data is first transferred entirely to cache storage from tape, and then transferred to the destination. Alternately, the data is first transferred entirely from the source to cache storage, and then written to tape. If no cache service is available, the job terminates.
direct (2)	-qos_direct_only	BOTH	The data is transferred immediately to the source as it is read from tape. Alternately, DIVA writes the data to tape immediately as it is transferred from a destination. If no direct transfer service is available, the job terminates.

Value (API Code)	Job Option	Job Type	Description
cache and direct (3)	-qos_cache_and_direct	BOTH	If cache transfer is not available, for example no Actor with cache storage is available, then a direct transfer is performed.
direct and cache (4)	-qos_direct_and_cache	BOTH	If a direct transfer is not available, for example no Actor with direct transfer enabled is available, then a cache transfer is performed.
nearline (5)	-qos_nearline_only	RESTORE	This is only available for Restore and N-Restore jobs. If a Nearline disk instance exists, the data is transferred from Nearline disk to the destination. Alternatively, the data is first transferred entirely to Nearline storage on disk from tape, and then transferred to the destination. If no Nearline service is available, the job terminates.
nearline and direct (6)	-qos_nearline_and_direct	RESTORE	This is only available for Restore and N-Restore jobs. If Nearline transfer is not available, for example no Actor with Nearline storage is available, then a direct transfer is performed.

If an object to be restored has both disk and tape instances, and Cache Only, or Cache and Direct QOS is used, DIVA may restore the tape instance as first priority over the disk instance. This behavior depends on the DIVA_CACHE_QOS_USE_DISK setting in the DIVA configuration. If set to *true*, DIVA restores the disk instance regardless of the QOS specified.

The Cache transfer method is particularly important for optimum use of DIVA resources when the transfer speeds between tape devices and the server vary considerably. For example, if the tape drive in the job can write data at 400 Mbps, but the source can only

deliver the data at 100 Mbps, the tape drive never achieves its optimum transfer rate. Using the Cache QOS, the file can be completely transferred to cache first, and the drive can complete its write operation at its maximum speed. This method enables the drive to be used for other jobs in a shorter time compared to the same transfer using the Direct QOS.

If an object to be restored has a disk instance the Nearline Only or Nearline and Direct QOS restores from that instance. If an object to be restored has only tape instances, the Nearline Only or Nearline and Direct QOS attempts to create a permanent disk instance and then restore from that instance. Every subsequent Nearline restore for the same object is blocked and waits for the first restore process to create a disk instance. If the first restore fails to create a disk instance the process repeats with the next restore attempting to create a disk instance. All other restores are blocked until the disk instance has been created.

The default QOS for Restore and N-Restore jobs is Nearline and Direct. If the restore job is a Transcode Restore, or if the destination server is a Movie2Me server, the Manager switches the restore QOS to Direct Only. Other QOS types aren't supported in this case.

Unmanaged Storage

This chapter describes source- and destination-server configurations for each type of unmanaged storage repository.

A source server is an unmanaged storage repository that is used as a source of files by an archive request. A destination is the equivalent for restore requests (partial restore and multi restore too).

Topics

- Unmanaged Storage Repositories
- General Parameters
- Avid MSS (Program Stream) Servers
- Avid Airspace Servers
- Supported -title and Third-Party IDs
- Avid Interplay Server
- SeaChange BMS and BMC Servers
- SeaChange BML Servers
- SeaChange BMLe and BMLex Servers
- Leitch vR Series Servers
- Leitch Nexio Servers
- Grass Valley Profile Servers
- Grass Valley UIM Gateway
- Grass Valley K2 Servers
- Grass Valley M-Series iVDR Servers
- Sony Optical Drives
- Sony Hyper Agent Servers
- Sony MAV70 Servers
- Omneon Spectrum MediaDirector Servers (QuickTime)
- Omneon MediaGrid Content Storage System
- Quantel Power Portal Gateway
- Standard FTP and SFTP Servers
- Local Source Servers

- Disk and CIFS Source Servers
- Configuring Cloud Object Storage as Unmanaged Servers (Azure, S3...)
- Expedat Servers from Data Expedition
- Metasources

Unmanaged Storage Repositories

Unmanaged storage repositories are independent platforms, products, and systems. Unmanaged storage repositories include broadcast video servers, FTP servers, disk storage arrays, object storage, and hard drives. DIVA manages content on a variety of unmanaged storage repositories.

A *source* unmanaged storage repository holds content to be transferred into DIVA. A *destination* unmanaged storage repository receives content from DIVA.

The configuration of source and destination repositories determines how DIVA issues jobs to them, and how DIVA manages two or more simultaneous jobs in the Jobs queue.

Defining an Unmanaged Storage Repository

You define unmanaged storage repositories that you want DIVA to manage and use in DIVA jobs. To define an unmanaged storage repository for DIVA, do the following:

1. To display the *UNMANAGED STORAGE* view, browse to *System Settings* > *Unmanaged Storage*.
2. Give each server type or disk file system an unique name, and configure as follows:

Source Only

DIVA archives only files from the server or disk file system.

Destination Only

DIVA restores only files to the server or disk file system.

Source and Destination

DIVA archives and restores files to and from the server or disk file system.

Common Job Parameters for Unmanaged Storage Repositories

The following parameters are common to all jobs involving source and destination unmanaged storage repositories on a destination server:

- The Source Type is the protocol or access method used when interacting with the target device.
- The maximum number of read and write transfer sessions and the total maximum number of read/write sessions combined. This identifies the limits on the number

of simultaneous jobs that DIVA executes concurrently on the target device, or prioritizing write (Restore) operations over read (Archive) operations.

- Define the maximum bandwidth allowable to DIVA for transfers to or from the device. This may be used to throttle data transfers where the target device is shared with other Networks or third party applications.
- The default Quality of Service (QOS). This is the QOS used when *Default* is specified in a job's Quality of Service field.
- Define Connect Options that must be provided (or that can also be optionally specified) for the specific protocol or access method of the target device. Examples of Connect Options are recursive sub-folders, user names or passwords, or other options specific to the selected source type. If no options are specified, DIVA ignores this parameter.
- Specify the Root Path to the files to be archived on the source, or restored to on a destination. Specify the Root Path as an absolute directory path on the target device. For example `c:\Exported\MPEG2`. The Root Path configuration also depends on the source type, and can be left blank in some cases (and is ignored by DIVA).

  For Local or Disk source types, the Root Path specifies the mount point of the device in the local file system.

The Connect Options and Root Path parameters defined for a server might not be appropriate for every job submitted. As a work-around, you can specify correct Connect Options and Root Path parameters at the job level. Whether a job overrides the server parameters depends on the source type.

The Files Path Root specified in a job can either be appended to the Root Path specified in the server configuration, or, if it's specified as an absolute path, it can override the root path entirely.

For a comprehensive list of options, paths, and interactions at the job level, see the *DIVA Installation and Administration Guide*, at https://www.telestream.net/telestream-support/diva/support.htm.

General Parameters

The following topics describe configuration parameters for features, connection options, and most servers.

Files Path Root Parameter

The Files Path Root (FPR) parameter is for Archive and Restore jobs. This parameter specifies the root folder for data transfers and applies to any type of server.

You can enter either an absolute or a relative path in the *Files Path Root* field. This parameter is limited to 32,767 characters.

Each content server topic specifies the expected format of the Files Path Root and related File Names parameters for Archive jobs.

For Partial File Restore jobs, the file names on the destination are those specified when archiving. If no Files Path Root is entered, DIVA uses the one specified during archiving.

Root Path Parameter

The Root Path is a server attribute that you can use as a default path for FTP-like servers, or as a disk-mount point for disk and local source servers. This applies to any type of server. The path is appended before any Files Path Root specified in jobs, unless the path specified in a job is an absolute path.

This approach provides better server abstraction. You specify the server directories used by DIVA at the configuration level, not at the job level. You can change the server directories at any time without changing to DIVA clients.

The Root Path value is always an absolute path defined by the operating system. An Omneon Path is the player name and is always considered an absolute path.

Absolute path names are supported on Windows to a maximum of 4000 characters. Relative path names are limited to 256 characters on Windows systems.

If you leave the Root Path field empty, DIVA ignores the parameter. However, if you do specify a Root Path, its value is combined with the Files Path Root you specified in a job to create the final server path. This process is performed according to the following rules:

- Relative paths are added to the absolute path, absolute paths override preceding absolute paths (standard Path Arithmetic).
- If the Root Path and Files Path Root have different operating system types, DIVA converts the second path (Files Path Root) to the operating-system type specified by the first path (Root Path) by replacing "\" with "/" (and vice versa). The converted path is then considered the relative path.
- If the Root Path ends with a ">," the Files Path Root is always considered to be a relative path, and DIVA omits the ">" during concatenation.

Server ROOT_PATH	**Object: Original_FPR recorded in database & metadata**	**Job Type**	**Files Path Root (FPR)**	**Resulting rule applied to create actual path for the transfer**	**Resulting path considered for the transfer**	**Resulting original Files Path Root (FPR) recorded in database and metadata**
Null		Archive	Null	ROOT_PATH +FPR	Null	Null
Null		Archive	Set	ROOT_PATH +FPR	FPR	FPR
Set		Archive	Null	ROOT_PATH +FPR	ROOT_PATH	Null
Set		Archive	Set	ROOT_PATH +FPR	ROOT_PATH+FPR	FPR
Null		Archive with tr_arch format	Null	ROOT_PATH +FPR	Null	Null
Null		Archive with tr_arch format	Set	ROOT_PATH +FPR	FPR	Null
Set		Archive with tr_arch format	Null	ROOT_PATH +FPR	ROOT_PATH	Null
Set		Archive with tr_arch format	Set	ROOT_PATH +FPR	ROOT_PATH+FPR	Null
Null	Null	Restore	Null	(ROOT_PATH+FPR) | Original_FPR	Null	
Null	Null	Restore	Set	(ROOT_PATH+FPR) | Original_FPR	FPR	

Server ROOT_PATH	**Object: Original_FPR recorded in database & metadata**	**Job Type**	**Files Path Root (FPR)**	**Resulting rule applied to create actual path for the transfer**	**Resulting path considered for the transfer**	**Resulting original Files Path Root (FPR) recorded in database and metadata**
Set	Null	Restore	Null	(ROOT_PATH+FPR) | Original_FPR	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 Archive				Null
	Set	Transcode Archive				Null

Style Paths

The following table describes Windows style paths for the Root Path, File Path Root, and the actual path to the files.

Root Path (Server)	**File Path Root (Job)**	**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`

Metasource Parameter

A Metasource server manages several servers that share the same online storage, or multiple Drop Folder Monitors. The Metasource parameter includes failover and load-balancing features, and can be applied to any type of server.

For more information on the Metasource server types, see *Metasources*.

Connect Options Parameter

Use the Connect Options parameter to specify the communication protocol with servers, or to modify the protocol defaults.

Some Connect Options specified for a server may be superseded by options specified in jobs. The following connect-option descriptions specify whether an option is superseded by a job parameter.

Archiving and Restoring Filename and Path Renaming Rules (`-arch_renaming`, `-rest_renaming`, `-arch_path_renaming`, `-rest_path_renaming`)

This feature is available for Archive and Restore jobs. 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.

Renaming rules are associated with servers. Configure filename or path renaming during archive or restore using the DIVA Web App. The configuration can be superseded by the job option.

Use these parameters when a workflow implementation requires automatic filename or path renaming during object archiving, when the object is (partially) restored, or when a transcoded object is re-archived or restored.

Rename files at archive time (`-arch_renaming`) or at restore time (`-rest_renaming`). Rename relative path at archive time (`-arch_path_renaming`) or at restore time (`-rest_path_renaming`).The format for these parameters are as follows:

```
-arch_renaming [renaming_rule]+
-rest_renaming [renaming_rule]+


-arch_path_renaming [renaming_rule]+
-rest_path_ renaming [renaming_rule]+


renaming_rule =
[activation_format:expression_patterns:output_format]
```

The `-arch_renaming` option enables renaming files during the archive process. This option is typically used for the following example cases:

- A file extension must be added to archived files.
- When associated to a transcoder cache (Local Server), archive renaming rules can be set to rename the files of a transcoded clip. This is useful when files created by the transcoder do not have the expected names.

The `-rest_renaming` option enables renaming of files during the restore process. This is typically used when the server requires strict naming of files, and the files being transferred do not follow these rules.

This option is available for Restore, Partial File Restore (this is an alternate way to rename partially restored files), and N-Restore. If multiple renaming rules are defined, DIVA processes the rule for each server independently.

The `-arch_path_renaming` and `-rest_path_renaming` options enable renaming relative paths for files at archive and restore time. The relative path to be renamed is not the Path Root, it is the relative path between the Path Root and the files.

At least one `renaming_rule` must be specified for the option. All renaming rules are located in the DIVA Web App except the Service Name and Port parameters. DIVA goes through each renaming_rule for each file on the list to be transferred starting with the first one:

- The rule is applied if the file name matches this rule's `activation_format`.
- The condition is satisfied if the beginning of a file name matches the evaluation condition of the first rule.

  For example, a condition such as .*\.track will be satisfied by all of the following file names—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) to check whether the renaming rule must apply. This is useful when renaming paths because the relative path of each file is checked using the activation format. For example, DIVA could rename the path of some files depending on file extensions.

The expression_patterns parse the file name. It is a regular expression, which will include up to nine special symbols to identify different parts of the file name: `\1 \2 \3 \4 \5 \6 \7 \8 \9`.

The output_format is an expression that qualifies the format of a renamed file, based on atomic items (\1 through \9) previously identified when applying expression_patterns to the original file name. Two additional specific symbols can be used:

- `\o` indicates the object name
- `\c` indicates the object collection

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/.

The following examples describe different possible scenarios and their associated outcomes using these parameters.

Example One

To add the GXF extension to all files archived from GVG Profile (by default, these files do not have an extension). If a file does have an extension, the GFX extension will not be added. Use the following statement to achieve this:

```
-arch_renaming <.*\..*:(.*)\.(.*):\1.\2><.*:(.*):\1.gxf>
```

This statement will process the file names as follows:

Input file Name	Output File Name
Star Wars	Star Wars.gxf
Readme.txt	Readme.txt
Jaws.gxf	Jaws.gxf

Example Two

Use the following statement to remove the .gxf extension (if any) at archive time:

```
-arch_renaming <.*\.gxf:(.*)\.(.*):\1>
```

This statement will process the file names as follows:

Input File Name	Output File Name
Star Wars.gxf	Star Wars
Readme.txt	Readme.txt
Jaws.avi	Jaws.avi

Example Three

When Vantage 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 aren't those expected by Pinnacle MSS servers, and this option fixes these discrepancies. Use the following statement to achieve this:

```
-arch_renaming
<.*\.header:(.*):header><.*\.ft:(.*):ft><.*\.info:(.*):info><.*MSS
:(.*):std>
```

This option will process the file names as follows:

Input File Name	Output File Name
`FOO.MSS.header`	header
`FOO.MSS.ft`	ft
`FOO.MSS.info`	info
`FOO.MSS`	std

Example Four

A variation of the following statement can be used to re-parent some files under a different relative directory:

```
arch_path_renaming <media:(.*):media.dir>
```

With this option all the files under media will be moved to media.dir.

To aid in regular expression development, regular expression exercisers are published at http://regexone.com/ and http://www.lornajane.net/posts/2011/simple-regular-expressions-by-example.

To use this feature, you should understand basic regular expression syntax. You can find Regular Expression introductory information online at http://www.hathitrust.org/, http://books.google.com/, and http://www.gutenberg.org/.

Connection Timeouts Option (`-list_timeout` | `-transfer_timeout` | `-control_timeout`)

These options specify the maximum timeout values allowed for different FTP connection operations, and override the default timeout settings. The timeout value for directory and file listings (`-list_timeout`), file transfers (`-transfer_timeout`), and control port connections (`-control_timeout`) can be set.

If an operation exceeds the set timeout value the operation is terminated.

The default value is used if a timeout parameter is not specified or if the timeout value is set to zero.

Options format:

```
-list_timeout {integer}
-transfer_timeout {integer}
-control_timeout {integer}
```

Where {integer} is the maximum allowed timeout in seconds.

Here are the default timeout values for each option:

Option	Default Timeout
`-list_timeout`	120 seconds
`-transfer_timeout`	180 seconds
`-control_timeout`	120 second

Deleting Source Server Content after Archiving (-allow_delete_on_source)

This parameter specifies if an Archive job can use the Delete on Source QOS option, and is in the format `-allow_delete_on_source`.

The Archive job optional parameter delete_on_source instructs DIVA to delete the original asset on the source server after the archive of the object is successfully completed.

If this option is specified in an Archive job and the source server Type is not LOCAL, DISK or CIFS, DIVA automatically terminates the job.

This parameter applies to the FTP_STANDARD source server type. you can change this behavior so that jobs will not fail when `delete_on_source` is specified in an Archive job.

If the `-allow_delete_on_source` option is specified, and the delete_on_source parameter is specified in an Archive job, DIVA will attempt to delete the asset from the source server after the archive has been completed successfully.

This option can't be superseded by the job option.

DIFWAV

The sequence is `clip.dif`, and then WAV files.

DIRS_FIRST

The sequence places directories first and is as follows:

```
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 sequence places files first and is as follows:

```
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;
```

This ensures that files are archived in the correct sequence so that they are restored in the correct sequence when restoring them to a real Pinnacle MSS server (a real Omneon server).

DPX Partial File Restore doesn't examine the file name or the DPX header information to determine which file is assigned to which frame. The assignment is based purely on the sequence in which the .dpx files appear within the archive. By default this sequence is based on ordering established by the source server, and is typically alphanumeric. For example, NTFS DISK servers order files and folders aren't case-sensitive as a general rule (but not where diacritical marks, such as ', `, or ^ are applied). By default, when DIVA encounters a sub-folder it recursively processes all of the children of that folder (including sub-folders) before continuing with other files. If a folder appears in the alphanumeric folder listing, it is archived recursively in the order it appears.

However, this can create some issues. A user may want all of the subdirectories of a given directory processed first followed by the files in the directory, or the user might want all files processed first followed by subdirectories. InDIVA, the actor allows the archive options `-file_order DIRS_FIRST` or `-file_order FILES_FIRST` to address these issues as previously described.

Example

An archive contains SeaChange SAF files. These files must be transcoded, and then restored to a Pinnacle MSS server. In this case, the LOCAL source server used by the transcoding process is defined with the -file_order MSS option (among others).

This ensures the files coming out of the transcoder are archived and restored in the correct sequence. That is, header, ft, info and then std.

FTP Socket Window Size (`-socket_window_size`)

This option specifies the total buffer space per data socket reserved for send and receive. This option applies to some server types using FTP protocol, such as FTP_STANDARD, OMNEON, PDR, or MSS.

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, DIVA uses the default value set at the operating system level. Telestream recommends increasing this value to 32768 or more on fast networks. Some performance tests must be run to identify the best setting.

```
-socket_window_size <bytes>
```

The {number} is the buffer size in bytes.

FTP Socket Block Size (`-socket_block_size`)

Note: The `-socket_bufsize` option is deprecated. Telestream recommends that you don't use it, because it might conflict with the `-socket_block_size` parameter.

This option defines how many kilobytes the actor tries to send and receive in a single system call during FTP transfers. For example, if the internal buffer size of the actor is set to 2 Mb and `-socket_block_size` is set to 64 KB, 32 system calls are required to write a single buffer to a data socket.

This option applies to some server types using FTP, such as FTP_STANDARD, OMNEON, PDR, MSS, and others.

Option format: `-socket_block_size` {integer}, where, {integer} is the buffer size in kilobytes, ranging from 32 to 2048 kilobytes.

FTP Passive Mode Transfers (`-pasv`)

The *-pasv* option specifies that the FTP data connection must be opened in passive mode (as opposed to active mode) for the associated server. This may be necessary if a firewall exists between the actor and the server.

This option applies to some server types using FTP, such as FTP_STANDARD, OMNEON, PDR, or MSS.

Ignoring Files During Restore

Use the following statement to ignore files during restore:

```
-rest_ignoring {<rule>} [<rule>|<rule>|<rule>]
```

Continue to add <rules> as necessary in the previous statement.

There are no predefined set of values for these options. Possible values for this option are infinite and fully customizable.

The files being 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 is restored and the job will be complete.

During an N-Restore, if multiple renaming rules are defined, DIVA will process the rule for each server independently.

Example

A typical use case is restoring a SeaChange clip to a destination that doesn't support SeaChange special files (private data and video index files). The following statement prevents a server from restoring files with PD or VIX extension:

```
-rest_ignoring <.*\.pd><.*\.vix>
```

The results if the previous statement are as follows:

DIVA Object	Destination Server
`Clipname.pd`	
`Clipname.vix`	
`Clipname`	Clipname

Metadata Restore Option (`-rest_metadata`, `-rm`)

This option specifies that a metadata file must be generated and restored on every Restore job. This option applies to any server type. Because video servers may reject the metadata file, this option actually applies mainly to LOCAL, DISK and FTP_STANDARD types.

Either form of the option can be used as follows:

```
-rest_metadata
```

```
-rm
```

When an object is restored, it is first restored normally. 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 job.

The metadata file format is compliant with the DIVA File Set Watch Folder Metadata File specification. The metadata file name is virtualobjectname.mdf.

MSS

The sequence is header, ft, info, and then std.

MSS Server in MXF Mode (`-mxf`)

This option specifically applies only to MSS server types, otherwise DIVA ignores it. You use this option to indicate when a MSS server is configured to import and export MXF wrapped clips.

There are no additional parameters for this option and you include it in the following format:

```
-mxf
```

Number of Actors to Retry Option (`-num_actors_to_retry`)

Use this option to limit the number of actors that an Archive, Restore, or Partial File Restore job will be retried on. By default, this option is not specified and there is no limit. Therefore, all actors will be tried in case the job constantly fails.

This option applies to any server type and can't be superseded by the job option.

This option uses the following statement:

```
-num_actors_to_retry {number}
```

The {number} is the number of retries to attempt and can include zero.

Example

The option `-num_actors_to_retry 3` means that the Manager performs no more than four operations (total) with different actors, even if there are more than four actors configured. That is, the initial job plus three retries for a total of four attempts.

OMNEON

The sequence is clip.mov, and then essence files (WAV, AIFF, M2V, MPEG, and DIFF).

Quality of Service (`qos=`)

This option specifies the transfer mode used when transferring from this specific server when the archive initiator sets the `QualityOfService` parameter in Archive or Restore parameters to DEFAULT.

This parameter applies to any type of server, and can't be superseded by the job option.

If the archive initiator sets the QualityOfService to something other than DEFAULT, DIVA ignores the `qos=Connect Option`.

The format for the parameter is `qos=[DIRECT_AND_CACHE|CACHE_AND_DIRECT]`.

Note: This option must be the first one in place in the Server Connect Options field, and must be specified in lowercase.

The QOS value is not case-sensitive and no longer must be specified at the beginning of the options. For example a valid option is: `-login test -pass test qos=nearline_only`.

NEARLINE_ONLY and NEARLINE_AND_DIRECT QOS values are supported in the Server Connect Options. These options are valid only for a Restore job. DIVA ignores the setting and the usual default is applied if a source or destination server with either setting is used in any other type of job.

DIRECT_AND_CACHE Direct transfers to or from a server, and to or from DIVA are preferred. However, cache transfers occur if processing the job in direct mode is not possible.

CACHE_AND_DIRECT Cache transfers to or from a server, and to or from DIVA are preferred. However, direct transfers occur if processing the job in cache mode is not possible.

See also *Server (Source/Destination) Connect Options*.

The following table describes sample Quality of Service connections:

QOS Connect Option	QOS Set by the Archive Initiator	Actual Transfer Mode Applied by the Manager
DIRECT_AND_CACHE	DEFAULT	DIRECT_AND_CACHE
DIRECT_AND_CACHE	DIRECT_ONLY	DIRECT_ONLY
DIRECT_AND_CACHE	CACHE_ONLY	CACHE_ONLY
CACHE_AND_DIRECT	DEFAULT	CACHE_AND_DIRECT
CACHE_AND_DIRECT	DIRECT_ONLY	DIRECT_ONLY
CACHE_AND_DIRECT	CACHE_ONLY	CACHE_ONLY
	DEFAULT	DEFAULT (that is, DIRECT_AND_CACHE)
	DIRECT_ONLY	DIRECT_ONLY
	CACHE_ONLY	CACHE_ONLY

Restore Ignore Option (`-rest_ignoring`)

This option is available for Restore, Partial File Restore, and N-Restore jobs. It ignores some files during restore based on one or more regular expression rules. The possibilities offered by regular expressions are versatile and enable many different types of filtering.

Files matching one of the regular expressions are ignored by the server. The rule supports Unicode characters to offer maximum flexibility.

Restore Ignore File Relative Paths Option (`-ignore_relative_path`)

If this option is specified on a source or destination server, DIVA will restore all the files of an object directly to the Files Path Root, ignoring any relative paths found in the list

of files. This option is useful when absolute paths were specified in the list of files when objects were archived.

Restore AXF Mode Option (`-axf`)

The `-axf` parameter is optional for Restore jobs and instructs the Manager to restore the original asset into an AXF File. Instead of restoring the content of an object to the destination, DIVA restores the content into a new AXF File.

Combined with the `-rm` or `-rmxl` parameters, you can use this option to export an object with metadata information and then drop it into a DFM watch folder.

This option applies to FTP_STANDARD, SFTP, LOCAL, DISK, and EXPEDAT server types.

Restore Specific Order Option (`-file_order`)

This option specifies the file order during archive, and also the file order during restores.

Use this option archiving or restoring files that are MSS files (Omneon QuickTime files), but the archiving source server is not an AVID (Pinnacle) MSS server (an Omneon server).

This option is not limited to specific server types, but it is only meaningful for LOCAL, DISK, CIFS, and FTP_STANDARD servers. This option can be superseded by the job option.

Specify the file sequence during archiving or restoring using the following statement:

```
-file_order {MSS|OMNEON|DIFWAV|SEACHANGE DIRS_FIRST|FILES_FIRST}
```

The following list identifies the archive sequence for specific formats:

SEACHANGE

The sequence is `clip.pd`, `clip.vix`, and then `clip`.

Server FTP User Log In (`-login`)

This option is generally used to specify a user name to connect to a server when the transfer protocol is FTP or FTP-like, and is in the format `-login {user_name}`.

This option applies when specified in server type description, and can be superseded by the job option.

Values applicable to a specific server type are detailed in the related topic.

Server CIFS User Log In for CIFS and FTP (`-user`)

This option is generally used to specify a user name to connect to a CIFS or FTP server, and is in the format `-user {user_name@domain}`. This option applies when specified in server type description, and can be superseded by the job option.

Possible values applicable to a specific server type are detailed in the related topic.

Server Password (`-pass`)

This option is generally used in combination with the -login option, and is in the format `-pass [password]`.

This option applies when specified in server type description, and can be superseded by the job option.

Possible values applicable to a specific server type are detailed in the related topic.

Server Connection Port (`-port`)

This option is used when a port parameter is required to connect to a server, and specifies the port number in the format `-port [port_number]`.

This is an integer value that applies when specified in server type description, and can be superseded by the job option.

Possible values applicable to a specific server type are detailed in the related topic.

Transcode Format Option (`-tr_archive_format`, `-tr_restore_format`)

Each factory in a transcoder determines the format of the output file. These options allow you to define the factory and output format.

They apply to any server type, and have no fixed list of values. This option can't be superseded by the job option unless used in a TranscodeArchived job.

These options specify the 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}
```

The {factory_name} is the name of the Vantage workflow.

Transcoder Name Option (`-tr_names`)

Use this option to specify the transcoder to use for transcode operations. It applies to any server type and can't be superseded by the job option, unless used in a TranscodeArchived job.

Either the `-tr_archive_format` or the `-tr_restore_format` option must always be used with `-tr_names`. When transcoding is applied, one of the transcoders defined by `-tr_names` is selected by DIVA according to the transcoders defined in the DIVA configuration based on the availability, configured queue size, and configured performance of the transcoder.

The format for this option is as follows:

```
-tr_names {transcoder_name} [transcoder_name]
```

The {`transcoder_name`} is the name of a DIVA Transcoder defined in the Transcoders area of the Systems view of the DIVA Web App.

See *For detailed configuration information, see the DIVA Installation and Administration Guide.*.

Using a Temporary Filename when Restoring to a Destination

By default, DIVA restores objects to a destination using the destination filename. There are situations where it could be desired to restore to a temporary file first, and then rename the temporary file at the end of the transfer.

To restore to a temporary file, and then rename the temporary file at the end of the transfer, do the following:

1. Add `-restore_to_temp_files` to the *Unmanaged Storage (source/destination) Options* field.

   This option is supported by LOCAL or SMB based disk Unmanaged Storage, and also FTP-based.

   Just before renaming a file to its final name, DIVA checks whether this file already exists. If it does exist, DIVA deletes it so that the renaming operation doesn't fail.

Avid MSS (Program Stream) Servers

Avid (previously Pinnacle) MSS Video servers can be installed in one of the following configurations:

Independent Storage

The video server 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 three files as listed below (or four if the optional information file is present). They are always archived and restored in the following sequence:

header—This is the first file and the clip's header.

ft—This is the second file and the frame table.

std—This is the third file and the video and audio essence.

info—When present, this is the fourth file. It is an optional information file.

All files are located in a folder that matches the name of the clip (that is, if the clip name is FOO, the files are located in a folder also named FOO).

Newer MediaStream servers can export and import clips with a MXF wrapper. When configured for MXF, the server generates a single file (std) which is the MXF file. DIVA archives only one STD file in MXF Mode. The file is automatically renamed to {clipname}.mxf. This mode is not supported by independent storage servers.

MSS with Independent Storage

One record is created for each MSS that DIVA must move data to and from.

Attribute	Value	Example
IP Address	MSS IP address	10.80.114.21
Source Server Type	MSS	MSS
Connect Options for Systems with One Gateway	`-login {gw_host_name}` `-pass .video_fs`	`-login fcgate1` `-pass .video_fs`
Connect Options for Systems with Two Gateways	`-login {gw1_host_name}[, gw2_host_name]` `-pass .video_fs`	`-login fcgate1,fcgate2` `-pass .video_fs`

In a system with two gateways, fcgate1 and fcgate2, DIVA manages failover between the two when a connect option such as `-login` fcgate1, fcgate2 is specified. If the initial FTP connection fails with fcgate1, it will be retried on fcgate2.

Note: This feature has been deprecated and is now implemented using the METASOURCE source server type.

MSS with Shared Storage

One record is created for each gateway connected to the storage network that DIVA must move data to and from.

Attribute	Value	Example
IP Address	IP Address of the gateway through which DIVA accesses the shared storage.	`10.80.114.28`
Source Server Type	MSS	MSS
Connect Options	-login video_fs -pass .video_fs	`-login video_fs` `-pass .video_fs`

MSS with Shared Storage in MXF Mode

One record is created for each gateway connected to the storage network DIVA has to move data to and from.

Attribute	Value	Example
IP Address	IP Address of gateway through which DIVA accesses shared storage.	`10.80.114.28`
Source Server Type	MSS	MSS
Connect Options	-login video_fs (or -login mxf_fs) -pass .video_fs (or -pass .mxf_fs) -mxf	`-login video_fs` `-pass .video_fs`

MSS with DIVA_archiveVirtualObject

The following table describes typical server example parameters.

Parameter	Value	Example
FilesPathRoot	The name of the clip.	CITIZENKANE
FileNames	*	*

Avid Airspace Servers

Avid Airspace (previously known as Pluto) is a video server with independent storage. Each clip deals with a single essence file located on the storage root. Airspace uses standard FTP protocol to transfer files to and from the video server internal storage over a Gigabit Ethernet Network.

One record is created for each video server DIVA has to move data to and from.

Attribute	Value	Example
IP Address	IP address of the video server.	`10.80.114.28`
Source Server Type	FTP_STANDARD	FTP_STANDARD

Connect Options	-login {FTP_user_name}
	-pass {FTP_password}
	-port {FTP_port_number}
		`-login ftpuser`
		`-pass Pa$$word`
		`-port 6530`

The following table describes an Avid Airspace server use example:

DIVA_archiveVirtualObject Parameter	Value	Example
FilesPathRoot	Leave this field empty.	
FileNames	Enter the name of the clip in this field.	TRAFFIC

Supported -title and Third-Party IDs

The following requests support the `-title` option:

- `Archive`
- `Restore`
- `Partial Restore`
- `Copy To Group`
- `Transcode Archive`
- `Delete Instance`
- `Copy Tape`

`-title` is a request option used to represent the *Third party ID* associated with a request. If specified in a request, it is stored alongside the request.

You should specify the title with the key word `title` prefixed with a dash and followed by a space, separated string in quotes ex: `-title` “a title.”

Alternatively, you can specify a title with a single word without any spaces. ex: `-title` oneWord.

The *Third party ID* is specified in the request options as `-title` "a title" or `-title` oneWord; for example: `-title` “Migrate Service Job Id 22” or `-title` JobId-22.

DIVA REST API supports `-title` for *Third Party ID* in DIVA Manager.

Avid Interplay Server

An Avid Interplay server can restore a master clip or a sequence to Avid Interplay. You can use this, for example, to restore clips or sequences that are not referenced by the Avid archive engine database.

For restores to complete successfully, the selected object must contain the AAF of the sequence (or master clip) and all the referenced media files.

Configuring an Avid Interplay Server

This setting allows an actor to form a URL to the workflow service REST API to request that an AAF is checked in to interplay. An actor does this after restoring something to the interplay.

This is an example of the gateway URL: `https://ipaddress:8765`. You can't use `127.0.0.1` here since the actor may not be on the same machine as the API gateway.

To add an Avid Interplay server, do the following:

1. In the DIVA Web App, browse to *Configuration* > *Resources* > *General Settings* > *REST API Gateway*.

   DIVA opens the *REST API GATEWAY* view.
2. Specify the *API Gateway URL*.
3. Click *Save All Changes*.

Adding an Avid Interplay Server

To add an Avid Interplay server, do the following:

1. Browse to *Configuration* > *Resources* > *System Settings* > *Unmanaged Storage*.
2. Click the Add Unmanaged Storage button.

   DIVA opens the *ADD SERVER* view.
3. Select *Avid Interplay* from the *Server Type* drop-down menu.

4. Leave the other fields on the *ADD SERVER* view blank, and click *Save*.
5. Install the Avid ISIS client on your DIVA server. Once it's installed, reboot your system.

   The the Avid ISIS client is located here: \\172.16.10.45\Divarel\Releases\04_AvidInterface\Avid_Isis_Direct\V3.0\Interplay_Client - Working.
6. Open the control panel for the Avid ISIS client. Right click each workspace, assign it a mount point (UNC), and check the AutoMount.
7. Reboot.

   You can now browse Avid ISIS storage in the DIVA Web App.

Restoring an Avid Interplay Profile

To restore a profile from an Avid Interplay server, do the following:

1. Browse to *Configuration* > *Resources* > *System Settings* > *Unmanaged Storage*.
2. Click the name of the Avid Interplay server.

   DIVA opens the *EDIT SERVER* view.
3. Click *ADVANCED SETTINGS*.

   DIVA opens the *EDIT SERVER ADVANCED SETTINGS* view.
4. Select the desired profile from the *Restore Profiles* drop-down menu, and click *Save*.

SeaChange BMS and BMC Servers

A SeaChange BMS (Broadcast Media Server) is a standalone video server equipped with a fast-Ethernet Interface and its own storage.

A SeaChange BMC (Broadcast Media Cluster) is a cluster of video servers providing unified storage based on SeaChange RAID2 technology. Each server of the BMC can deliver files stored on RAID2 to DIVA using FTP. The file transfer format is SeaChange Archive Format (SAF) only.

Note: The SeaChange FTP servers do not support directories. All files must be listed under the FTP root directory.

By default, a SeaChange BMC node offers Automatic Load Balancing management for data transfer across all nodes of the cluster.

To use this feature, declare the last node of the BMC in the DIVA configuration. In this case, DIVA always connects to the same node of the cluster. This node transparently redirects transfers to other nodes as required.

You can disable SeaChange Automatic Load Balancing with a special IP address setting in the DIVA configuration. In this case, declare all nodes of the BMC in the DIVA configuration. For more information, see *IP Address* in the following table.

You can also add a Metasource that encompasses all nodes of the cluster to enable load balancing and failover from within DIVA.

Attribute	Value	Example
IP Address	IP address of the BMS or BMC node. You can disable the SeaChange Automatic Load Balancing by placing a $ in front of the IP address of all BMC nodes. The syntax for this is $IP_Address.	`10.80.114.26` $10.80.114.26
Source Type	SEACHANGE_BMC	SEACHANGE_BMC
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

SeaChange uses a flat file system. Specify these parameters to archive a clip.

DIVA_archiveVirtualObject Parameter	Value	Example
FilePathRoot	Leave this field empty	
FileNames	Enter the name of the clip in this field.	POKEMON

SeaChange BML Servers

The SeaChange BML (Broadcast Media Library) is a large storage system for SAF (SeaChange Archive Format) files and is based on the RAID2 technology of the SeaChange BMC platform.

A SeaChange BMC (Broadcast Media Cluster) is a cluster of video servers providing unified storage based on SeaChange RAID2 technology. Each server of the BMC can deliver files stored on RAID2 to DIVA using the FTP protocol.

DIVA uses the FTP protocol to communicate with either a BMS or BMC. You can only overwrite the files when the Actor service is stopped. The file transfer format is SAF (SeaChange Archive Format) only.

Note: The SeaChange FTP servers do not support directories. All files must be listed under the FTP root directory.

The Automatic Load Balancing feature as described for BMC also exists for BML and operates in a similar fashion.

Attribute	Value	Example
IP Address	IP address of the BML Node. You can disable the SeaChange Automatic Load Balancing by placing a $ in front of the IP address of all BMC nodes. The syntax for this is $IP_Address.	`10.80.114.26` $10.80.114.26
Source Server 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 (enabled) and 0 (disabled). The default value is 1 (enabled).

SeaChange BML clip storage is flat. Specify these to archive a clip:

DIVA_archiveVirtualObject Parameter	Value	Example
FilesPathRoot	Leave this field empty.	
FileNames	Enter the name of the clip in this field.	OFFICESPACE

SeaChange BMLe and BMLex Servers

The SeaChange BMLe is the storage subsystem of the latest SeaChange MediaClient architecture. SeaChange BMLe is 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 earlier IOP interfaces. Each node of the cluster is equipped with four FSI ports to provide high speed transfers to and from the BMLe and BMLex.

DIVA uses CIFS or FTP protocols to communicate with BMLe and BMLex.

File transfer format is the native format of the files stored on the BMLe and BMLex. Each asset consists 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 consists of three files (that is, the essence file, VIX, and PD), the files are always archived and restored by DIVA in the following sequence:

PD

This is the private data file and the first file archived or restored.

VIX

This is the index file and the second file archived or restored.

Essence File

There is no extension on this file and it is the last one archived or restored.

DIVA can restore SAF (SeaChange Archive Format) files from the archive to the BMLe or BMLex. When a SAF clip is restored to a BMLe or BMLex, the SAF file is automatically unwrapped by DIVA and the three files are restored to BMLe or BMLex (that is, the essence file, .pd file, and .vix file). This server can also restore SAF files from an archived SAF object to BMLe.

This feature is transparent to you because DIVA automatically detects SAF and unwraps it in real time. When a SAF clip is restored to the BMLe, the SAF file is unwrapped by DIVA and the name of each file is extracted from the SAF file header. The content is restored to BMLe in the separate files previously described.

BMLe and BMLex generated files support SAF releases SAF 0.1, SAF 1.0, and SAF. SAF may contain two consecutive private data files including a 12 byte .pd file, and a 28 byte .pd file. In this case, DIVA will only restore the 28 byte file while ignoring the 12 byte file.

Declare one server for each FSI of each node:

Attribute	Value	Example
IP Address	IP address FSI	`10.80.114.26`

Attribute	Value	Example
Source Server Type	SEACHANGE_BML	SEACHANGE_BML

Connect Options	`-ftp or -cifs`
	`-login {FTP_user_name}`
	`-user {cifs_user_name@domain}`
	`-pass {password}`
	`-nometadata`
		`-cifs`
		`-user me@ourdomain.com`
		`-pass Pa$$word`

`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 or -cifs

One of these two options must be specified. Otherwise, Streaming API protocol is assumed, which is not supported by DIVA for BMLe and BMLex. This option can't be superseded by the job option.

-ftp

FTP protocol is used for data transfer to and from BMLe and BMLex.

-cifs

CIFS protocol is used for data transfer to and from the BMLe and BMLex FSI cards. The implicit CIFS path to BMLe is \\fsi_ip_address\vstrm.

-nometadata

This option prevents DIVA from archiving the .vix and .pd files when the clip being transferred includes essence, .vix, and .pd files. This option can't be superseded by the job option.

Specify these parameters to archive a clip:

DIVA_archiveVirtualObject Parameter	Value	Example
FilesPathRoot	Leave this field empty.	
FileNames	Enter the name of the clip in this field.	HANNITY

Leitch vR Series Servers

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, Archive Streamer gateway. The Archive Streamer uses FTP protocol over a Gigabit Ethernet network.

Note: The Leitch vR Source Type is depreciated. It was initially created to follow the first Archive Streamer releases that did not correctly report the size of the file to be transferred.

A record is required for each Archive Streamer server you are using in DIVA.

Attribute	Value	Example
IP Address	IP address of Leitch Archive Streamer	`10.80.114.21`
Source Type	FTP_STANDARD	FTP_STANDARD

Connect Options	-login {FTP_user_name}
	-pass {FTP_password}
	-port {FTP_port}
		`-login ftpuser`
		`-pass Pa$$word`
		`-port 6021`

Specify these parameters to archive a clip:

DIVA_archiveVirtualObject Parameter	Value	Example
FilesPathRoot	Leave this field empty.	
FileNames	Enter the name of the clip in this field.	FRIENDS

Leitch Nexio Servers

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.

To move clips in and out of the shared storage is possible directly from the video server using the standard FTP protocol over a Gigabit Ethernet network.

Note: The Leitch Nexio Source Type is deprecated.

A record is required for each Nexio server you are using in DIVA.

Attribute	Value	Example
IP Address	IP address of Leitch Nexio video server.	`10.80.114.21`
Source Server Type	FTP_STANDARD	FTP_STANDARD

Connect Options	-login {FTP_user_name}
	-pass {FTP_password}
	-port {FTP_port}
		`-login ftpuser`
		`-pass Pa$$word`
		`-port 6021`

Specify these parameters to archiving a clip:

DIVA_archiveVirtualObject Parameter	Value	Example
FilesPathRoot	Leave this field empty.	
FileNames	Enter the name of the clip in this field.	ENEMIES

Grass Valley Profile Servers

Grass Valley Profile video servers are provided in one of two ways; with independent storage, where the video server includes its own fault tolerant disk storage, or as part of a MAN, where video servers are connected to a SAN where the fault tolerant disk storage resides.

Irrespective of the storage mechanism, the actor always connects to a specific Profile server. The exchange format is GXF only.

Profile Storage consists of one master disk (for example, EXT: or INT1:), and one level of folders where one clip is seen as one file. One folder called *default* always exists.

The network infrastructure between GVG Profiles and actors is an IP/FC network.

A record is required for each GVG Profile server you are using in DIVA.

Attribute	Value	Example
IP Address	IP address of the video server.	`10.80.114.21`
Source Server Type	PDR	PDR
Name	Logical name for the video server.	GVG-Profile-1

The actor configuration parameters are located in the actor area of DIVA. The two parameters in the following table directly influence transfer performance. Technical Support recommends trying several value combinations on the target platform.

In addition to these two parameters, the MTU size setting for the HBA used for IP/FC traffic to the Profile servers may also have an influence on transfer performance.

Grass Valley doesn't provide any recommendation for MTU size. However, Technical Support recommends setting the MTU size on the actor HBA to the same value as the MTU size of the Profile HBA. This is only a recommended setting and not an absolute rule.

Attribute	Description	Recommended Values
DIVAACTOR_PROFILEREADINGBS	The FTP block size (in bytes) used for transfers on Profile video servers in reading.	1500 16374 32768 (default)
DIVAACTOR_PROFILEWRITINGBS	FTP block size (in bytes) used for transfers on profile video servers in writing.	16374 32768 (default)

Specify these parameters when archiving a clip:

DIVA_archiveVirtual Object Parameter	Value	Example
FilesPathRoot	/explodedFile/disk:/folder	/explodedFile/INT1:/default
FileNames	Enter the name of the clip in this field.	MyClip

Grass Valley UIM Gateway

UIM is a gateway to standalone or MAN Grass Valley Profile servers. It provides TCP/IP over Gigabit Ethernet connections to external systems (such as DIVA). For legacy purposes, the connection can also be IP/FC for regular profiles.

UIM also provides real-time format conversion (to MXF). The UIM exchange format is GXF (by default), or alternately MXF.

A record is required for each UIM gateway you are using in DIVA.

Attribute	Value	Example
IP Address	IP address for the UIM.	`10.80.114.21`
Source Server Type	PDR	PDR

Connect Options	`-login {movie|mxfmovie}`
	`-format {?D10AES3}`
	`-extension {file_extension}`
		`-login mxfmovie`
		`-format ?D10AES3`
		`-extension .mxf`

-login

Specifies the FTP user for logging onto the UIM to achieve transfers in the desired format. The two available logins are movie (for GXF exchange format), and mxfmovie (for MXF exchange format). The movie user is assumed if -login is not specified.

-format

The UIM supported options for some file formats. This depends on -login option. The only available option is ?D10AES3. The ?D10AES3 option is an e-VTR compliant file format used with the -login mxfmovie option. If this option is not specified, MXF files will be processed in Grass Valley OP1a format. This option is not specified by default.

This option can be superseded by the job option.

-extension

This option adds the specified extension to the original clip name in the archive. For example, if the original clip is clip1 and the -extension .mxf option is specified, then the archived file will be clip1.mxf.

You should suppress the specified extension before restoring to the destination if it already exists. For example, if the archived file is `clip1.mxf` and the `-extension .mxf` option is specified, the restored file on the destination will be named `clip1`, without an extension.

This option is deprecated and replaced by the *-arch_renaming* and the *-rest_renaming* options. This option can be superseded by the job option.

UIM are gateways to the Profile server. Use this the same way for UIM and Profile servers regardless of the transfer format (GXF or MXF).

DIVA_archiveVirtual Object Parameter	Value	Example
FilesPathRoot	/explodedFile/disk:/folder	/explodedFile/INT1:/default
FileNames	Enter the name of the clip in this field.	MyClip

Grass Valley K2 Servers

To DIVA, K2 servers are similar to Profiles and UIM combined. K2 servers offer Gigabit Ethernet connections to external systems, and the exchange format is GXF (default), and alternately MXF.

A record is required for each K2 server you are using in DIVA.

Attribute	Value	Example
IP Address	IP address of the K2 server.	`10.80.114.21`
Source Server Type	PDR	PDR

Connect Options	-k2
	-login {movie|mxfmovie}
	-format {?D10AES3}
	-extension {file_extension}
		`-k2`
		`-login mxfmovie`
		`-format ?D10AES3`
		`-extension .mxf`

-k2

This specifies the interface with the K2 servers. When this option is set, DIVA will retrieve the size of the file to be transferred before the actual archive transfer (K2 FTP does support the SIZE command). Correct transfer progress is reported by DIVA.

When this option is not set, DIVA will assume that servers are Profile, and will not retrieve the file size before archive transfers. Progress will then remain at 0% before suddenly jumping to 100% when the transfer is complete.

This option has no impact on transferred content, and can be superseded by the job option.

-login

This option specifies the FTP user for logging onto the K2 server to achieve transfers in the desired format. The two available logins are movie (for GXF exchange format), and mxfmovie (for MXF exchange format). The movie user is assumed if -login is not specified.

-format

The K2 supported options for some file formats. This depends on -login option. The only available option is ?D10AES3. The ?D10AES3 option is an e-VTR compliant file format used with the -login mxfmovie option. If this option is not specified, MXF files will be processed in Grass Valley OP1a format. This option is not specified by default, and can be superseded by the job option.

-extension

This option adds the specified extension to the original clip name in the archive. For example, if the original clip is clip1 and the -extension .mxf option is specified, then the archived file will be clip1.mxf.

If the specified extension already exists it must be suppressed before restoring to the destination. For 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 replaced by the `-arch_renaming` and the `-rest_renaming` options. This option can be superseded by the job option.

You use this the same way for K2 and Profile servers regardless of the transfer format (GXF or MXF).

DIVA_archiveVirtual Object Parameter	Value	Example
FilesPathRoot	/explodedFile/disk:/folder	`/explodedFile/INT1:/default`
FileNames	Enter the name of the clip in this field.	MyClip

Grass Valley M-Series iVDR Servers

Grass Valley iVDR is an analog and digital VTR that includes a Gigabit connection for material exchange of GXF files. The iVDR exchange protocol is similar to the exchange protocol for Profile servers.

A record is required for each iVDR server you are using in DIVA.

Attribute	Value	Example
IP Address	IP address of the iVDR.	`10.80.114.21`
Source Server Type	PDR	PDR
Name	Logical name for the iVDR.	GVG-iVDR

Specify these parameters to archive a clip:

DIVA_archiveVirtual Object Parameter	Value	Example
FilesPathRoot	/explodedFile/disk:/folder	/explodedFile/INT1:/default
FileNames	Enter the name of the clip in this field.	MyClip

Sony Optical Drives

For Sony Optical Drives, whether a media is Write-Once can be viewed by clicking on the tape (the Write-Once property is displayed in the Tape Properties window). The Blu-ray discs are shown as tapes and viewable in the Tapes view. The Write Once Media column displays this information as either Y or N indicating whether the tape is Write-Once. The view can also be filtered so only Write-Once media is displayed.

Sony Hyper Agent Servers

Hyper Agent is the name given to the Newsbase FTP server from Sony. The implementation of this FTP server is specific because the LIST command returns a proprietary formatted list of files. This list contains duration, and start and 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 jobs. The progress bar is not affected by the approximated size.

Duration, frame rate and bitrate are retrieved using the following two commands, which are set by the actor at the beginning of each Archive job:

SITE FSIZ {Clip ID}

This SITE command returns the duration of the specified clip.

SITE GCNF

This SITE command returns the current system configuration of the server. This system configuration must remain the same to ensure that all of the clips on the server are the same.

The following example is a log entry of communications between actor and the Hyper Agent FTP:

Note: In the following log example the word configuration is misspelled; this is a bug in the FTP server and appears in logs as shown in the example.

SITE FSIZ 1444247

200 150 (the duration is 150 frames)

SITE GCNF

213-System configuration

PAL (the frame rate is 25 frames per second)

20

30.0 (the Bitrate is 30 Mbps)

D10

SD_IFRAMEONLY

213 End of system configuration

A record is required for each ClipBox server you are using in DIVA.

Attribute	Value	Example
IP Address	IP address of the Newsbase server.	`10.80.114.21`
Source Server Type	SONY_HYPER_AGENT	SONY_HYPER_AGENT
Connect Options	`-login {user_name}` `-pass {password}`	`-login sony` `-pass sony`

DIVA_archiveVirtualObject Parameter	Value	Example
FilesPathRoot	Leave this field empty.	
FileNames	Enter the Clip ID in this field.	1444247

Sony MAV70 Servers

The Sony MAV70 video server has its own independent storage. MAV70 storage organization is flat and all files reside in the storage root.

A record is required for each MAV70 server you are using in DIVA.

Attributes	Value	Example
IP Address	IP address of the MAV70 server.	`10.80.114.21`
Source Server Type	FTP_STANDARD	FTP_STANDARD
Connect Options	`-login {user_name}` `-pass {password}`	`-login wing` `-pass mpegworld`

Specify these parameters when archiving a clip:

DIVA_archiveVirtual Object Parameter	Value	Example
FilesPathRoot	Leave this field empty.	
FileNames	Enter the name of the clip in this field.	MyClipName

Omneon Spectrum MediaDirector Servers (QuickTime)

The Omneon MediaDirector is the heart of the Omneon Spectrum architecture. It is connected to MediaPorts or MultiPorts which handle isochronous ingest and playback, and to external storage that is usually shared with other Omneon MediaDirectors.

You can use either MediaStore or MediaGrid for external storage. This topic describes connecting MediaDirector to MediaStore storage for MediaGrid support in DIVA.

DIVA 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. The actors use unique FTP site commands for smart and transparent access to essence files (in particular, the automatic discovery of a folders structure and collision-avoidance at restore time).

A record is required for each MediaDirector server you are using in DIVA.

Attribute	Value	Example
IP Address	IP address of Omneon Director.	`10.80.114.21`
Source Server Type	OMNEON	OMNEON
Root Path	Either leave this field empty or enter an absolute clip directory.	/default/clip.dir

Connect 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, DIVA restores the QuickTime reference file in the following sequence:

1. Audio Tracks

2. QuickTime File

3. Video track

The restore workflow is specific when this option is set. DIVA uses the temporary folder to cache the QuickTime file.

-tempdir_mode

This option performs a Partial File Restore of MXF files, and is applicable only to Omneon servers. The MXF Partial File Restore job will terminate if this option is not included in the job.

DIVA_archiveVirtual Object Parameter	Value	Example
FilesPathRoot	Enter the absolute clip directory in this field, or leave this field empty to use the configured Root Path.	`/default/clip.dir`
FileNames	Enter the name of the clip in this field.	MyClip

Omneon MediaGrid Content Storage System

MediaGrid is the Content Storage System from Omneon to which Omneon Spectrum servers can be connected.

The MediaGrid system consists of two major components; ContentServers that store and provide access to media, and ContentDirectors that act as overall file system controllers. ContentDirectors manage the distribution of data throughout the system.

Like any other client system, DIVA gets access to the media through a MediaGrid ContentDirector. DIVA 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 FSD (File System Driver), installed on MediaGrid clients hides this complexity to client systems.

Note: Install the Omneon FSD on the server where the actor service is installed.

You can download Omneon FSD for Windows at http://support.omneon.com/Updates/Omneon/Current/MediaGrid/WinFSD. Use the default password for the site (if required).

When material is in QuickTime format, the essence files are stored using a specific folder structure. DIVA 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).

Use UNC paths to access SMB servers, and managed disks if the UNC path is directly mounted on the actors.

When the actor is running as a Windows service, MediaGrid shares are accessed through a UNC path because drive letters mapped to network drives aren't accessible by Windows services. In this case ensure the following:

- Omneon MediaGrid folders being accessed by DIVA are properly shared for a given Windows user.
- The Actor service is configured to run under this user account.
- The user has local administrative rights on the actor.

A record is required for each ContentDirector server you are using in DIVA.

Attribute	Value	Example
IP Address	Leave this field empty.	
Source Server Type	MEDIAGRID	MEDIAGRID
Root Path	`\\ContentDirector\filesystem\clip.dir`	`\\10.30.0.200\cldev4\clip.dir` `\\mycontentdir\fs5\clip.dir`

DIVA_archiveVirtual Object Parameter	Value	Example
FilesPathRoot	Leave this field empty.	
FileNames	Enter the name of the clip in this field.	MyClip

In cases where the asset is wrapped as QuickTime, DIVA searches for files matching the format `clipname.mov` or `clipname.MOV`. DIVA automatically retrieves and processes all of the potentially referenced files.

In cases where the material is wrapped as MXF, DIVA searches for a file matching the format `clipname.mxf` or `clipname.MXF`. There is only one file per clip.

Quantel Power Portal Gateway

The *Quantel Power Portal* was previously referred to as the *ISA Gateway*. An ISA system consists of these components:

- **ISA Manager**—The ISA Manager contains the Clip Database. Clips are identified using a unique FID (File Identifier) in the ISA System.
- **Q or sQ Servers**—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. For example, sQ server ID 1 contains POOL ID 1 and POOL ID 2, sQ server ID 2 contains POOL ID 3, and sQ server 3 contains POOL ID 4.
- **Power Portal** (ISA Gateway)—an FTP server that imports and exports clips.

A record is required for each Power Portal server you are using in DIVA.

One record must be created for each Power Portal.

Attribute	Value	Example
IP Address	IP address of the video server.	`10.80.114.21`
Source Server Type	QUANTEL_ISA	QUANTEL_ISA
Connect Options	`-login {FTP_user_name}` `-pass {FTP_password}`	`-login ftpuser` `-pass Pa$$word`

The configuration parameters are located in the DIVA actor area.

Parameter	Description	Suggested Values
DIVAACTOR_QUANTELRENAMECLIPS	Enables and disables the file renaming feature.	0 indicates the renaming feature is disabled. 1 indicates the renaming feature is enabled.

`DIVAACTOR_QUANTELRENAMECLIPS` applies to restore jobs only. If this parameter is set to 1, and the object name format is `clipName,UID` (this is Omnibus naming), then object-related files are renamed using clipName as the Name Root.

For example, if the object Superman,01AB45 is composed of files 8152.D10 and 8152.WAV, and is restored to a QUANTEL_ISA destination, the following is true:

- If `DIVAACTOR_QUANTELRENAMECLIPS` is set to 0 (disabled), DIVA transfers files called 8152.D10 and 8152.WAV to Power Portal.
- If `DIVAACTOR_QUANTELRENAMECLIPS` is set to 1 (enabled), DIVA transfers files called Superman.D10 and Superman.WAV to Power Portal.

Quantel storage is a flat structure. Specify these parameters to archive a clip:

DIVA_archiveVirtualObject Parameter	Value	Example
FilesPathRoot	Leave this field empty.	
FileNames	FID1.ext1[,FID1.ext2,]	`clip.mxf,clip1.tar`

Files coming from Power Portal can be different file types including: D10+WAV (file names similar to 8152.D10 and 8152.WAV), MXF (TestClip.mxf), and TAR (FramesDifference.tar).

If a file is restored twice to Power Portal, the first file is not overwritten. The second restore creates a new file that is identified by a new FID. The actor captures the new FID after the transfer and forwards it to the Manager.

Your program must call *DIVA_GetRequestInfo* to obtain the new FID using the DIVA API. If the job completes, the new FID is in the job's `ADDITIONAL_INFO` element within ClipID tags. The ClipID tag is encapsulated in the `ADDITIONAL_INFO` element:

```
<ADDITIONAL_INFO>
 <ClipID>8546</ClipID>
</ADDITIONAL_INFO>
```

Automation is also free to specify a POOL ID in the FilePathRoot Restore job parameter. If no POOL ID is specified, Power Portal automatically assigns one at restore time.

Standard FTP and SFTP Servers

DIVA can interface with any standard FTP server or SFTP servers (known as SSH FTP or Secure FTP).Video servers supporting a fully RFC-959 compliant FTP server are considered standard FTP servers. You set this parameter in the Home Directory panel of the FTP Site Properties for Microsoft IIS FTP servers.

Create one record for each video server that DIVA transfers data with.

Attribute	Value	Example
IP Address	IP address of the FTP server.	`10.80.114.21`
Source Server Type	FTP_STANDARD or SFTP	FTP_STANDARD

Connect Options	`-login {user_name}`
	`-pass {password}`
	`-port {port_number}`
		`-login moon`
		`-pass mars`
		`-port 27`

-login

This is the FTP or SFTP user name. The default value is anonymous.

-pass

This is the FTP or SFTP user's associated password. The default value is anonymous.

-port

This is the port number the FTP or SFTP server is listening on for connections. The default value for FTP_STANDARD is 21, and for SFTP is 22.

You can specify parameters three different ways for archive jobs as described in the following table:

DIVA_archiveVirtualObject Parameter	Value	Example
`FilesPathRoot`	Full path to files Partial path to files No path entry	/my_videos/movies /my_videos
`FileNames`	Names of files Partial path and names of files Full path and names of files	maniolia, matrix movies/maniolia, movies/matrix /my_videos/movies/maniolia, /my_videos/movies/matrix

DISK_FTP_PASSIVE_MODE

By default, data connections are created in active mode. In active mode, the DivaFtp client connects from a random, unprivileged port that is higher than port 1023. Then, it starts listening on the port and sends a PORT command to the FTP server. Valid values for this parameter are 0 (disabled) and 1 (enabled).

When `DISK_FTP_PASSIVE_MODE` is set to 1 (enabled), data connections are created in passive mode. In passive mode, DivaFTP sends a PASV command and the server (not the client) creates the socket.

DISK_FTP_BLOCK_SIZE

The `DISK_FTP_BLOCK_SIZE` parameter defines how much data the actor tries to send and receive with a single system call during FTP transfers. For example, if the internal buffer size of the actor is set to 2 MB 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

The `DISK_FTP_SOCKET_WINDOW_SIZE` parameter adjusts 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 bytes.

Local Source Servers

A local source server represents a disk partition for a specific actor (internal disks, NAS or SAN disks), and is tied to a specific actor (versus a disk source server not tied to any particular actor).

One record must be created for each local source server DIVA must transfer data to and from.

Attribute	Value	Example
Name	Enter the same name as the actor this source server is bound to.	actor1
IP Address	Enter the same IP address as the actor this source server is bound to.	`10.80.114.21`
Source Server Type	LOCAL	LOCAL

You can specify archive job parameter values using three different patterns:

DIVA_archiveVirtual Object Parameter	Value	Examples
FilesPathRoot	Full path to files Partial path to files No path entry	`/my_videos/movies` `/my_videos`

FileNames	Names of files
	Partial path & file names
	Full path & file names
		`maniolia, matrix`
		`movies/maniolia, movies/matrix`
		`/my_videos/movies/maniolia`
		`/my_videos/movies/matrix`

If NT drive letters (for example, `E:`) are used, Telestream highly recommends leaving them in the FilesPathRoot section (that is, use scheme 1 or 2 in the previous table). Including them in the FileNames section prevents the job from replacing them with another path at restore time. Therefore, these objects can't be restored on a different platform.

Disk and CIFS Source Servers

A DISK or CIFS source server represents a disk partition assumed to be visible from all Network actors. The only difference between DISK and CIFS is the way blocks of data are read and written:

- DISK instructs actors to use (Windows) Direct I/O.
- CIFS instructs actors to use (Windows) Buffered I/O.
- Both DISK and CIFS source servers support UNC paths.

You should create one record for each DISK or CIFS source server DIVA has to move data from the source server to the destination server. You can also create a record for a generic source server to represent any type of DISK or CIFS source server.

Attribute	Value	Example
Name	Nickname of the source server.	`generic-disk`
IP Address	IP address of the source server.	`10.80.114.21`
Source Server Type	DISK or CIFS	`DISK`

You can specify archive job parameter values using three different patterns

DIVA_archiveVirtual Object Parameter	Value	Examples
FilesPathRoot	Full path to files Partial path to files No path entry	`/my_videos/movies` `/my_videos`

FileNames	Names of files
	Partial path & file names
	Full path & file names
		`maniolia, matrix`
		`movies/maniolia, movies/matrix`
		`/my_videos/movies/maniolia`
		`/my_videos/movies/matrix`

If NT drive letters (for example E:) are used, Technical Support highly recommends leaving them in the FilesPathRoot section (that is, use scheme 1 or 2 in the previous table). Including them in the FileNames section prevents the job from replacing them with another path at restore time. Therefore, you can't restore these objects on a different platform, for example on a Linux-based FTP server where drive letters aren't considered valid paths.

Configuring Cloud Object Storage as Unmanaged Servers (Azure, S3...)

You can declare Azure, S3 ACI and Google cloud object storage in DIVA as unmanaged servers.

Do the following:

1. First, in the cloud storage configuration, create a cloud account.
2. Browse to *Configuration* > *Cloud Storage*.
3. Where the name of the cloud account is specified in the address field of the unmanaged server, use the parameters of the cloud account in the unmanaged server.

See also *Cloud Storage Setup*.

Expedat Servers from Data Expedition

DIVA can interface with DataExpedition Expedat servers (up to release 1.17), also known as *servedat*. This solution uses MTP, which is a high performance file transfer protocol. This WAN acceleration software can use 100 percent of the bandwidth of any long distance or high latency networks.

This server works similar to the FTP_STANDARD server in terms of the FilesPathRoot and list of files.

When Expedat Server is configured with folders having the RestrictHome setting enabled, the RootPath for the Data Expedition Server entry must not reference an absolute path. The RootPath may be interpreted as a path that is not accessible from the Expedat home directory. For example, the Root Path / is interpreted as `C:\`. However, if the Expedat home directory is `D:\folder`, Expedat tries to access the path `D:\folder` on `C:\`, which is not valid. If the home directory is `C:\folder`, using the Root Path / is acceptable.

Instead of using an absolute path, use relative-path addressing to resolve this problem. For relative-path addressing leave the Root Path field empty in the DIVA Web App. Alternatively, for the archive or restore operation, specify the relative path in the FilesPathRoot field of the GUI Manager or API job.

To set up a default home location so that an API job can always use a file path, the Expedat `cv_password.txt` file must contain a login account assigned to a folder with the RestrictHome option set.

For example:

```
diva:diva:::S:\DFM:RestrictHome

diva1:diva:::S:\DFM1:RestrictHome

diva2:diva:::S:\some_other_folder:RestrictHome
```

The separate user login and password accounts allow for the creation of more than one EXPEDAT Server entry with different home locations. The API job can then reference the EXPEDAT Server pointing to the desired home location.

For detailed information on installation and configuration, see the *DataExpedition Expedat Server Installation Manual*.

Attribute	Value	Example
IP Address	IP address of the Expedat server.	`10.80.114.21`
Source Server Type	EXPEDAT	EXPEDAT

Connect Options	`-login {user_name}`
	`-pass {password}`
	`-port {port_number}`
	`-license {license_code}`
	`-encryption`
	`-seq_buffer_size {size_in_megabytes}`
	`-exp_maxrate {size_in_kilobytes}`
	`-exp_mindatagram {size_in_bytes}`
		`-login moon`
		`-pass mars`
		`-port 8080`
		`-license 46FE464A98`
		`-encryption`
		`-seq_buffer_size 16`
		`-exp_maxrate 1024`
		`-exp_mindatagram 2848`

-login and -pass

These options are mandatory if the server is configured with authentication parameters.

-port

This option should always be present because there is no default value.

-license

This is a mandatory parameter to use the DIVA Expedat Client. Without the license code the EXPEDAT Server is unusable. You can only configure one Expedat license key per Network.

-encryption

This option works with the Expedat Server, is optional, and enables Expedat content encryption during transfers.

-seq_buffer_size {size_in_megabytes}

This option defines the size of the DataExpedition internal buffer for each transfer. The default value is 16 MB 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. However, a small buffer consumes less memory.

-exp_maxrate {size_in_kilobytes}

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 {size_in_bytes}

This transfer protocol is over UDP. This option defines a minimum size for each network datagram payload that DataExpedition sends. The purpose is to prevent DataExpedition from sending too small of a packet over the network. You may want 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 can greatly reduce CPU overhead. However, using this setting without Jumbo Frames being fully supported can cause severe performance issues or loss of connectivity.

Metasources

A Metasource is a collection of multiple (single) source servers of the same type, where all source servers of a specific metasource share the same online storage. Each source server of the metasource should be of the same regular type (that is, any type except METASOURCE)—that is, the Metasource Base Type. A metasource provides load-balancing and failover mechanisms across all single source servers of the metasource.

The metasource source type allows several currently defined DIVA servers sharing the same online storage (or monitoring the same folder or FTP server for Drop Folder Monitors) to be combined and considered a single DIVA server configuration. This unique (and optional) feature enables DIVA to provide automatic load balancing and fail-over capabilities in case of one or more of the individual servers going offline.

When jobs are issued to DIVA with a server using the metasource source type, each additional archive or restore job uses the next server in the metasource list. If the server selected by DIVA is offline or encounters an error, DIVA automatically attempts to use the next server in the metasource list. If all servers fail to fulfill the transfer, the job terminates.

One record must be created for each Metasource DIVA has to transfer data to and from.

Attribute	Value	Example	Comments
Name	Name for video server's shared storage.	gvg-man-production	
IP Address	server1 [,server2,server3]	10.158.1.10,10.2.5.60,97.64.52.3	server1, server2,server3 must also be defined in the configuration as regular source servers of the same type (all types except METASOURCE, LOCAL, and DISK are permitted, for example, OMNEON, or PDR.
Source Server Type	METASOURCE	METASOURCE	
Network	Must be the same for Metasource and all single source servers.		Manager will not start if there is no match.
Site	Either one or the other of the sites from Metasource single source servers.		Site specified for Metasource is considered by Manager for resource selection.
Root Path	You can specify a Root Path at the Metasource level.		If the Metasource Root Path is null, the Root Path from the selected single source server is considered.
Max Accesses Max Write Acc. Max Read Acc. Max Throughput	Actual value for Metasource doesn't matter.		The value from the selected single source server is considered. You can't leave these fields empty. Technical Support suggests setting traffic regulation parameters to the sum of all single source server's respective parameters. Technical Support also recommends that you do not make any changes to this parameter while there are active jobs being processed because it can lead to job termination.
Connect Options	-failover_time={time_in_milliseconds} -retry_actor={number_of_retries}	-failover_time=300 -retry_actor=3	

-failover_time={time_in_milliseconds}

When a single source server is selected to process a job and it fails, the single source server is temporarily not considered part of the Metasource for 600 milliseconds.

This default value can be changed using this option. This option can't be superseded by the job option.

-retry_actor={number_of_retries}

Use this option to specify the number of Metasource single source servers to be tried for each actor that can be part of the job processing. The default, when this option is not specified, is 2.

For example, if the Metasource is defined as sd1, sd2, sd3, the set of possible actors is a1, a2, and -retry_actor is set to 2, DIVA will try a maximum of four combinations; most likely a0-sd1, a0-sd2, a1-sd3, a1-sd1.

This option can't be superseded by the job option.

Other single source server connection options can also be specified for the Metasource. The following table indicates the effects for the `qos= option` when specified at the Metasource level:

Option	Considered?	Description
`-login`	No	Value from selected single source server is considered. Applicable to FTP servers.
`-user`	No	Value from selected single source server is considered. Applicable to CIFS servers.
`-pass`	No	Value from selected single source server is considered.
`-port`	No	Value from selected single source server is considered.
`-allow_delete_on_source`	No	Implicitly assumed to be true if all single source servers (implicitly or explicitly) allow deleting on source server. Otherwise, assumed to be false.
`-arch_renaming`	No	Value from selected single source server is considered.
`-rest_renaming`	No	Value from selected single source server is considered.
`-file_order`	No	Value from selected single source server is considered.
`-tr_archive_format`	Yes	Values specified for single source servers do not matter.
`-tr_restore_format`	Yes	Values specified for single source servers do not matter.

Option	Considered?	Description
`-tr_names`	Yes	Values specified for single source servers do not matter.
`-rest_metadata`	No	Value from selected single source server is considered.
`-num_actors_retry=`	Yes	Values specified for single source servers do not matter.
`-ftp`	No	Value from selected single source server is considered.
`-cifs`	No	Value from selected single source server is considered.
`-nometadata`	No	Value from selected single source server is considered.
`-format`	No	Value from selected single source server is considered.
`-extension`	No	Value from selected single source server is considered.
`-k2`	No	Value from selected single source server is considered.

A Metasource is used the same as any source server of Metasource Base Type.

There are instances where it is required to delete content, and possibly the parent folder, on a server. To satisfy all possible scenarios there are two options available:

- `-r` deletes recursively
- `-delete_fpr` includes deletion of the parent folder

The two options, -r and -delete_fpr, work either separately or together, as described in the following workflow examples:

FilesPathRoot	Files	Options	Result
`C:\sourceserver\root`	*	`-r`	DIVA deletes the content of `C:\sourceserver\root` recursively.
`C:\sourceserver\root`	*	`-r -delete_fpr`	DIVA deletes the content of `C:\sourceserver\root` recursively, and then deletes root.
`C:\sourceserver\root`	*		DIVA deletes the content of `C:\sourceserver\root` (first level only).

FilesPathRoot	Files	Options	Result
`C:\sourceserver\root`	*	`-delete_fpr`	DIVA deletes the content of `C:\sourceserver\root` (first level only), and then eventually deletes root if it is empty.
`C:\sourceserver\root`	obj*	`-r`	DIVA deletes the content of `C:\sourceserver\root\obj` recursively, and then deletes `C:\sourceserver\root\obj`.
`C:\sourceserver\root`	obj*	`-r -delete_fpr`	DIVA deletes the content of `C:\sourceserver\root\obj` recursively, then deletes `C:\sourceserver\root\obj`, and finally deletes `C:\sourceserver\root` if it is empty.

FilesPathRoot	Files	Options	Result
`C:\sourceserver\root`	obj1* obj2*	`-r`	DIVA deletes the content of `C:\sourceserver\root\obj1` recursively, then deletes `C:\sourceserver\root\obj1`, and then deletes the content of `C:\sourceserver\root\obj2` recursively, and finally deletes `C:\sourceserver\root\obj2`.
`C:\sourceserver\root`	obj1* obj2*	`-r -delete_fpr`	DIVA deletes the content of `C:\sourceserver\root\obj1` recursively, then deletes `C:\sourceserver\root\obj1`, then deletes the content of `C:\sourceserver\root\obj2` recursively, then deletes `C:\sourceserver\root\obj2`, and finally deletes `C:\sourceserver\root` if it is empty.
`C:\sourceserver\root`	obj1* obj2\subfolder\clip.mov	`-r -delete_fpr`	DIVA deletes the content of `C:\sourceserver\root\obj1` recursively, then deletes `C:\sourceserver\root\obj1`, then deletes the content of `C:\sourceserver\root\obj2\subfolder\clip.mov`, then deletes `C:\sourceserver\root\obj2\subfolder` if it is empty, and then deletes `C:\sourceserver\root\obj2` if it is empty, and finally deletes `C:\sourceserver\root` if it is empty.

Media Storage Formats

This topic discusses the media formats supported by DIVA.

Topics

- Storage-Media Formats
- Tape Storage-Media Format
- Disk Storage-Media Format
- Object Instances Media Format
- AXF Objects
- DIVA Complex Objects
- DIVA Connect Complex Object WAN Transfers

Storage-Media Formats

In DIVA, a tape group, disk array, or cloud bucket has a Media Format parameter that determines which storage-media format DIVA employs when creating new archived objects. You can set he Media Format either to DIVA Legacy format or to one of the AXF formats. You can change this setting at any time. A change of this setting doesn't influence content already stored. Therefore, you can have more than one storage-media format within tape groups and disk arrays.

DIVA writes an object instance in one, and only one, media format. Therefore, if an object spans tapes, each tape used as part of an object instance is written in the same media format. In DIVA, an object can contain multiple instances, each of which can be stored in either Legacy format or one of the AXF formats.

Complex objects must be stored in AXF format. Because all complex objects are written in the AXF format, every instance of a complex object is in the AXF format.

Tape Storage-Media Format

Although a tape group can contain more than one storage format, an individual tape has (at most) one storage media format. DIVA assigns the tape-media format to an empty tape when it writes the first object to that tape. The tape is assigned the format of the tape group that appears in the job. After the media format for a tape is assigned, it can't be changed unless all objects on the tape are deleted. After deletion of all

objects from a tape, the tape's format becomes unassigned until DIVA writes content to the tape again. The tape format can't change unless it is empty and cleared.

Both Legacy- and AXF-formatted tapes can exist in the same tape group. Objects in AXF format are written only to AXF formatted tapes. Objects in Legacy format are written only to Legacy-formatted tapes even though they are in the same tape group.

A repack job always writes the destination tape in the same media format as the source tape. Similarly, tape spanning operations always uses the same format across all tapes storing spanned objects.

Disk Storage-Media Format

DIVA allows storing objects in different media formats on the same disk. If a disk containing objects in Legacy format is assigned to an AXF formatted array, the disk retains the objects in Legacy format. However, new objects written to the disk are written in AXF format.

Object Instances Media Format

Every tape and disk object instance is assigned a format of Legacy or AXF. The format of a tape or disk instance is assigned when the instance is created and is the format of the tape on which the instance resides. All instances on a tape must have the same 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 tape group specified in the job.

Tape groups or arrays used by complex object jobs must be in an AXF format because complex objects can't be stored in Legacy format. Therefore, any instance of a complex object is in the AXF format.

A migration job must be used to change a tape format from Legacy to AXF. Repacking a tape doesn't change the tape format. Repacking existing Legacy-format objects retains the format of the tape even if the tape group format was updated in the configuration from Legacy to AXF.

AXF Objects

Archive eXchange Format (AXF) is an open format that supports interoperability among disparate content storage systems and ensures the content's long-term availability regardless of how storage or file system technology evolves.

An AXF virtual 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 your valuable 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.

Native File and Folder Support

Users can see their files and folders in native format on archive devices rather than as an AXF container file. You can also access files and folders on storage devices like object storage. This access allows third-party software to perform operations, such metadata collection, face recognition, and transcodings, on the archive.

The following list identifies the different AXF formats available:

- **LTFS_AXF_1.1**

  This new format offers the same features as AXF_1.1 on an LTFS formatted tape and is only supported on tape. This format is not recommended for complex objects because it would generate very large LTFS indexes.

  The tape block size must be set to 524288 Bytes or greater; LTFS doesn't support lower block sizes.

  If an LTFS tape is loaded by LTFS software on a standalone drive, the contents of that tape can be accessed using Windows Explorer.

Caution: Access LTFS tape content in Windows Explorer only for recovery purposes. The LTFS software must not be running concurrently with DIVA on the same drives.

  The following features are supported by LTFS AXF:

  – Spanning

  – Drive compression

  – Drive encryption is supported, but not recommended if there is no procedure to export and load encryption keys to the LTFS software.

- **AXF_RF_1.1**

  This format uses the AXF 1.1 structures, but AXF files won't contain any overhead. This format is supported on disk arrays and object storage only.

Note: Telestream doesn't recommend using AXF_RF_1.1 format with complex objects.

  This format allows a user to see the files of an object on disk or cloud. When archiving complex objects with small files, performances are better using AXF_1.1 because the files are wrapped into an AXF container (this is the reason for the previous the note). When this format is chosen for a cloud storage array, there are limitations on the server side to the size of file that can be created. This limitation is approximately 5TB per file. Telestream recommends that you use AXF_1.1 instead of AXF_RF_1.1 if a DIVA object contains a file larger than 5TB.

- **AXF_1.1**

  This format is compliant with AXF 1.1 standards and is the recommended format when archiving complex objects. DIVA creates a new AXF segment every 500GB to avoid compromising multi-part upload performance when this format is selected

for cloud storage. Some object storage vendors limit the number of parts per object to 10000. The 500GB AXF segment limit maintains a reasonable size for each part.

- **AXF_1.0**

  This format is compliant with AXF 1.0 standards.

- **AXF**

  This is redirected to AXF_1.1.

- **LEGACY**

  This is the formal archive format used by DIVA (index.txt, 00000001, 00000002)

DIVA Complex Objects

A Complex Object stores more information about the files and folders in an archive, such as subtotals for each directory.

When you archive an object, DIVA determines whether the object should be complex or non-complex based on the number of component files in the object. The Complex Object Threshold sets the minimum number of components for making the object complex. The Complex Object Threshold default value is 5,000.

If the number of components is greater than the Complex Object Threshold value, DIVA archives a complex object. Otherwise, the object remains non-complex. A complex object remains a complex object in any context— even if you copy it using the Copy As command, or import it using the Export/Import Utility.

You can configure this threshold in the diva.conf configuration file. Telestream recommends leaving the threshold at the default value (5000 components) unless there is a specific reason to adjust the value.

When the Metadata Database feature is enabled, the Complex Object feature is available. When using Complex Objects, DIVA can track significantly more than the 10,000 files-per-object limit set for non-complex objects. The actual amount scales with system processing power and storage capacity.

A complex object differs from a non-complex object in some key ways. For example, the file and folder metadata information of a complex object is stored in the Metadata Database for Complex Objects not in the Postgres database. The file contains the file names, folder names, checksums, and files sizes. The directory that contains these files is the Metadata Database Root Directory. You must store a complex object in AXF format on either tape, disk or cloud.

A complex object maintains information about the folders and files in the archive. Complex objects store subtotals for each folder, including the total number of files and subdirectories within the folder, and the total size of all files within the folder and any sub-folders.

A complex object can contain hundreds of thousands of files. However, DIVA displays only a single placeholder file representing the complex object in Object Properties and Tapes dialog boxes.

See also *Complex Object*.

DIVA Connect Complex Object WAN Transfers

DIVA provides optional WAN acceleration that enables it to take full advantage of long distance, high latency, network paths (such as private site to site links or the public Internet), and can perform transfers of complex objects efficiently using Data Expedition MTP/IP protocol.

Example

1. DIVA instance 1 restores the complex object to the DIVA 2 system by first creating a new AXF file on the DIVA 2 system's Data Expedition server.
2. DIVA 1 restores all of the files from the local storage to the new AXF file created on the DIVA 2 Data Expedition server.
3. The DIVA 2 system creates a new AXF file on the destination (tape, disk, and so on).
4. The DIVA 2 system archives all of the files from the Data Expedition AXF file (created by DIVA 1 on the Data Expedition server) to the newly created AXF file on the destination.

   For more information, see the *DIVA Connect Installation, Configuration, and Operations Guide* at https://www.telestream.net/telestream-support/content-conductor/support.htm.

Data Management on Managed Disks

- Arrays, Disks, and Cache
- Creating a Disk Array
- Adding a Disk to a Disk Array
- Adding an Actor to a Disk
- Notifying the Manager and SPM

Managed disk refers to a local disk, a network share, a SAN disk or a NAS with a file system. This chapter contains information specific to configuration and operational aspects of Managed Disk storage and cache.

Arrays, Disks, and Cache

DIVA uses hard disk drive technologies for both the storage of Virtual Objects and for transient disk-cache storage during data transfers.

Any disk that DIVA uses is assigned to an array, a logical association of one or more disks for object storage. Disks configured as cache disks also are assigned to an array, typically named `CACHE`.

The storage of an object on a disk in DIVA is identified by the array name rather than by an individual disk name. DIVA automatically allocates objects among two or more disks within any array.

You can connect each disk in an array to a DIVA system, either directly in an actor host's hardware, as NAS; or through a SAN using Fiber Channel. With SAN, if the disk is to be accessed simultaneously from multiple actors, it can also employ additional file-system sharing software on the hosts.

You can configure individual disks in an array, as follows:

***Storage Only*.** The disk is only used for storage of objects. These types of disks employ some level of RAID technology to ensure data redundancy and protection against individual hard-disk failure.

***Storage and Cache*.** The disk is used for the storage of objects and also for caching operations. Both types use separate sub-folders on the disk. These types of disks

employ some level of RAID technology to ensure data redundancy and protection against individual hard-disk failure.

***Cache Only*.** The disk is used only for caching, tape-to-tape copying, tape spanning, or tape repacking operations. These types of disks may employ RAID technology to improve performance (for example, RAID 0).

***Storage and Nearline*.** The disk is used for the storage of objects, and also for Nearline operations. Both types use the same sub-folder on the disk. These types of disks employ some level of RAID technology to ensure data redundancy and protection against individual hard disk failure.

***Cache and Storage and Nearline*.** The disk is used for the storage of objects, caching, and Nearline operations. Both storage and Nearline types use the same sub-folder on the disk. However, the cache type uses a separate sub-folder. These types of disks employ some level of RAID technology to ensure data redundancy and protection against individual hard-disk failure.

With DIVA, you can configure individual disks for Read-Write access, Read-Only access, or you can disable them temporarily.

Caution: The file system of DIVA-managed disks should never be manipulated directly by any file, by DIVA, nor by a utility such as Windows Explorer. Moving, renaming, or deleting the folder structures or files can cause DIVA to mark that disk as Offline.
Using such a utility in any fashion completely destroys the disk's file system.

Disks that have file sharing software installed to provide shared host access--SNFS or MetaSAN, for example-- can appear as an unknown file system or as not being initialized to utilities such as Windows Disk Manager.

For information on configuring EMC ECS components, including object stores in arrays, actor-disk connections, array priority, object storage accounts, and validation checks, see the *DIVA Installation and Administration Guide*.

Creating a Disk Array

You can create a disk array with the DIVA Web App.

To create a disk array for archiving from DIVA, do the following:

1. In the DIVA Web App, browse to *Configuration* > *Disk Storage*.
2. Click *Arrays*.
3. Click the *Add Array* button (+).

   DIVA opens the *ADD ARRAY* view.
4. Enter a name and description.

5. Select a format.

   Telestream recommends AXF 1.1 or AXF RF 1.1 for new arrays.

   Don't select *Flashnet Loose* or *Flashnet Dtool*. These formats are for migrating from Flashnet to DIVA.

   See AXF Objects.
6. When the format is AXF RF 1.1, select a value from the *Object Path Prefix* drop-down menu. *Collection**Name* is commonly used.
7. Enter a value for *Max Allowed Disk Percent For Repack*.
8. To have DIVA read back and verify after archiving, enable *Verify Write*.
9. To set the level of priority of this disk array compared with the other disk arrays, enter a value into the *Priority* field.
10. Select a value from the *Archive QOS* drop-down menu.

    See *Quality of Service*.
11. Select a value from the *Restore QOS* drop-down menu.

    See *Quality of Service*.
12. To enable the array to access a disk via SMB, enable *Use Random SMB Access*.
13. Select a value from the *SPM Watermark* drop-down menu.

    The tooltip for the *SPM Watermark* field describes the values.
14. Click *Save*.

Adding a Disk to a Disk Array

You can add one or more disks to a disk array. To add a disk, do the following:

1. In the DIVA Web App, browse to *Configuration* > *Disk Storage*.
2. Click *Arrays*.
3. From the list of arrays, select the array to which you want to add a disk.
4. Click the Add Disk (+) button to the right of the array name.

   DIVA opens the *ADD DISK* view.
5. Enter a name for the disk.
6. Select a disk site from the *Site* drop-down menu.
7. Select a status from the *Status* drop-down menu.

   The Status is set to online by default. DIVA doesn't use offline disks.
8. Enter a value, in megabytes, into the Min Free Space field.

   DIVA keeps this amount of space on the disk free, so that archiving doesn't use 100% of the space available on the disk.
9. Select a value from the Verify Write drop-down menu.

   By default, DIVA inherits this value from the disk array. However, you can override the default. To have DIVA read back and verify after archiving, enable *Verify Write*. Otherwise disable *Verify Write*.

10. Click *Save*.

Adding an Actor to a Disk

You can add one or more actors to a disk. To add an actor, do the following:

1. In the DIVA Web App, browse to *Configuration* > *Disk Storage*.
2. Click *Arrays*.
3. From the list of arrays, select the array containing the disk to which you want to add an actor.
4. From the list of disks, select the disk to which you want to add an actor.
5. Click the Add Actor Connection (+) button to the right of the disk name.

   DIVA opens the *ADD ACTOR CONNECTION* view.
6. Select one or more actors from the *Actors* drop-down menu.
7. Enter a value for Max Throughput Mb/s.
8. Select a value from the *Used For* drop-down menu.

   For information about the types of usage, see *Arrays, Disks, and Cache*.
9. Select a value from the *Access* drop-down menu.
10. Enter the path, FTP address, or IP address, for the mount-point folder into the *Mount Point* field.
11. Select a value from the *Interface* drop-down menu.
12. Click *Save*.

Notifying the Manager and SPM

After creating arrays, adding disks, or adding actors, you can notify the Manager, notify SPM, or both. To send the notifications, do the following:

1. In the DIVA Web App, click the *Notify* button.

   See *Notify Button*.
2. To notify the Manager, select *Notify Manager*, and click *NOTIFY*.
3. Click the *Notify* button.
4. To notify SPM, select *Notify SPM*, and click *NOTIFY*.

See also *Creating a Disk Array*, *Adding a Disk to a Disk Array*, and *Adding an Actor to a Disk*.

Data Management on Object Storage

DIVA supports four main types of object storage: S3, Google Cloud Storage, Microsoft Azure, and Oracle Cloud Infrastructure.

Topics

- Cloud Storage Setup
- Target Media Archives
- Supported Object Storage implementations

Cloud Storage Setup

This section describes the setups for cloud accounts, cloud buckets, and actors.

See also *Configuring Cloud Object Storage as Unmanaged Servers (Azure, S3...)*.

Adding a Cloud Account

To create a cloud account, do the following in the DIVA Web App:

1. Browse to *Configuration* > *Resources* > *Cloud Storage* > *Cloud Accounts*.
2. Click the *Add Cloud Account* (+) button.

   DIVA opens the *CLOUD ACCOUNTS* view.
3. Enter a name for the cloud account, select the account type from the *Type* drop-down menu, and enter the data for the cloud account into the required fields.
4. Click *Save*.

   The DIVA Web App saves the setup information for the cloud account.

Creating a Cloud Bucket

To create a cloud bucket, do the following in the DIVA Web App:

1. Browse to *Configuration* > *Resources* > *Cloud Storage* > *Cloud Buckets*.

   DIVA opens the *CLOUD BUCKETS* view.

2. Click the *Add Cloud Bucket* (+) button.

   DIVA opens the ADD *CLOUD BUCKET* view.

3. Enter a name for the cloud BUCKET and enter, or select, the data for the bucket into the required fields.
4. Click *Save*.

   The DIVA Web App saves the setup information for the cloud bucket.

Connecting Actors to Cloud Buckets

To connect an actor to a cloud bucket, do the following in the DIVA Web App:

1. Browse to *Configuration* > *Resources* > *Cloud Storage* > *Cloud Buckets*.

   DIVA opens the *CLOUD BUCKETS* view.

2. From the list of cloud buckets, select the cloud bucket desired for an actor.
3. Click the *Add Actor Connection* (+) button.

   DIVA opens the *ADD ACTOR CONNECTION* DIALOG.

4. Enter data into the required fields, and click Save.

   The DIVA Web App saves the setup information for the cloud bucket.

Target Media Archives

For content that doesn't exist on the target media, DIVA performs a Copy to Tape Group to make an instance available on the target media. If the target media is an object storage, there is an optional parameter to specify the desired storage class.

If the content is available in archive-type storage, such as Glacier, DIVA stages the content to the target storage class. The number of days in which the content is available in the target storage class following a stage job is be configurable. The staging job may include a parameter to specify the number of days a restored object must be available. By default it is set to one day.

If the content is on a cloud bucket in archive-type storage such as Glacier, and the content has already been restored, DIVA checks the restoration expiry date and may eventually extend it if the number of days available requested is beyond the expiry date.

For validation purposes, the restoration expiry date can be verified on the S3 console to validate the functionality.

Supported Object Storage implementations

This section describes the setups for Object Storage implementations supported by DIVA.

S3-Based Object Storage (Amazon S3 Integration)

DIVA supports Amazon S3 AWS integration. Storage accounts allow a user to configure programmatic access to a user's AWS account. The configuration data in a DIVA storage account is exclusively used by DIVA actors to query S3 storage and transfer content to and from S3 buckets.

Upon creation of a storage account, the Configuration Utility automatically creates the set of DIVA resources needed to store DIVA managed objects in S3. The resources generated by the Configuration Utility are an Array, Disk, and actor-disk connections.

Amazon S3 Bucket Limitations

A given AWS account can't create more than 100 buckets. This limit can be optionally increased to 1000. Therefore, the number of objects that DIVA can archive per bucket has been increased to 100,000 AXF objects per bucket to enable more DIVA instances per cloud account.

Note: You can configure DIVA to write an unlimited number of objects in a given bucket replace by specifying a bucket name in the cloud bucket configuration.

When DIVA is functional, the configuration used by the Manager and other DIVA Modules is read from configuration files and DIVA database upon startup.

To make configuration changes, one needs to use -except when otherwise mentioned - the DIVA Web App. These changes are persisted in the corresponding configuration files, or DIVA database.

In order to force Manager or SPM modules to read the updated configuration parameters, the DIVA Web App provides Notification feature. It is a specific warning button in the top right of the web interface.

Other modules (BKS, Proxies, MDS...) are notified of the updated configuration parameters as soon as the configuration changes are saved in the DIVA Web App interface.

This chapter describes the changes to the DIVA configuration that become effective while the Manager is running, those changes that require a software component, and those changes which require you to restart the Manager.

Cloudian

DIVA includes support for Cloudian integration as both Storage Accounts and servers. Storage accounts allow users to configure programmatic access to a user's account. The configuration data in a DIVA storage account is used by the DIVA actors to query storage and transfer content. Storage Class supported = STANDARD.

EMC Elastic Cloud Storage (ECS) Integration

Instances stored on EMC Elastic Cloud Storage are local instances whose priority is lower than other types of local disk instances, but a higher priority than tape storage instances.

A DIVA Oracle Storage Class and Storage Location can be defined separately. If you require new cloud or local arrays in the future, you can specify all of these parameters as options. However, in DIVA both SWIFT and S3 are supported for interfacing with EMC ECS, but you can't change the existing configuration after the Array is configured.

You can set the Media Priority of a source instance for a Restore, Partial File Restore, and Copy to Tape Group jobs, which enables you to restore an instance stored on a local non-EMC ECS array with a higher priority than an instance on an EMC ECS array. If the priorities for the media are all the same, then the DIVA decides which source instance is preferred during these jobs.

For more information, see the *DIVA Installation and Administration Guide*.

Alibaba OSS

DIVA includes support for Alibaba OSS integration as both Storage Accounts and servers. Storage accounts allow users to configure programmatic access to a user account. The configuration data in a DIVA storage account is used by the DIVA actors to query storage and transfer content.

For more information on storage and servers, see the *DIVA Installation and Administration Guide*.

NetApp StorageGrid

DIVA includes support for NetApp StorageGrid integration as both Storage Accounts and servers. Storage accounts allow users to configure programmatic access to a user account. The configuration data in a DIVA storage account is used by the DIVA actors to query storage and transfer content. Storage Class supported = STANDARD.

Scality Zenko

DIVA provides support for Scality Zenko integration as both Storage Accounts and servers. Storage accounts allow users to configure programmatic access to a user's account. The configuration data in a DIVA storage account is used by the DIVA actors to query storage and transfer content. Storage Class supported = STANDARD.

S3 Server-side Encryption

In order to configure S3 server-side encryption, you can add these options to the Storage options field.

Option	Description
`-server_side_encryption=<KMS or AES256>`	You must set this option in order to enable S3 server-side encryption. DIVA supports SSE-S3, SSE-OSS and KSM. Set this option to AES256 if using SSE-S3 or SSE-OSS. Set this option to KMS if using KMS is required.
`-kms_key_id=<kms key ID>`	If `-sse` is set to KMS, set this option if DIVA should use a specific encryption key identified by its key id. This option is supported by aws:s3 and alib-aba:oss.
`-bucket_key_enabled` (aws:s3 specific)	If `-sse` is set to KMS, `-bucket-key-enabled` can be set if the KMS key ID to be used is associated with the bucket. This option is supported by aws:s3 but not alib-aba:oss. With Alibaba, there is no need to configure server side encryption on the DIVA side if the bucket is configured with a default server-side encryption.

Note: The `-kms-key-id` and `-bucket-key-enabled` parameters are mutually exclusive. If both are specified by mistake, DIVA will use `-kms-key-id` and ignore `-bucket-key-enabled`.

SSE-S3 or SSE-KMS Bucket Policy

When using SSE, the SSE option must be set to AES256 or aws:kms. Telestream recommends you add a bucket policy to prevent uploads without valid encryption headers.

When there is a bucket policy preventing uploads without encryption, or bad encryption header, the S3 server returns an HTTP error 403. Here is the error message returned by the DIVA actor when it happens:

```
Error during disk instance closure divastorage_GREG_STD_000002
[StorageCloudAPI error [16ba7b81-9dad-12d1-80b4-10c040243241.axf
upload failed, http response code: 403 [AccessDenied: Access
Denied]]]
```

Google Cloud Storage

Google Cloud Storage (GCS) is the object storage service offered by Google to Google Cloud account owners. Like any other object storage service, GCS offers an interface to create buckets and objects under buckets.

To set up Google Cloud Storage, first create a service account, then configure DIVA for Google Cloud Storage.

Creating a Service Account

You must associate DIVA with a service account, and with a specific role and permission. For security reasons, DIVA can't use the object storage interface in a regular Google account for configuring DIVA for Google Cloud Storage. Instead, you must create a service account via the Google console.

To create a service account, do the following:

1. Browse to the Google console at this location https://console.cloud.google.com.
2. Browse to *AIM & Admin* > *Service Accounts*.

   The Google console opens the *Create service account* dialog.
3. From the Role drop-down menu, select *Storage Admin*.
4. Click *Continue* > *Continue*.
5. Click *CREATE KEY* > *Done*.

   The Google console sends you a JSON key.

   The following table provides descriptions of the JSON key elements.

XML Element Under DeviceDisk	Description	Example
`DiskPath`	Object storage URL	https://www.googleapis.com
`Login`	This is set to the client email address found in the JSON key.	<name>@gcs-integration-to-diva.iam.gserviceaccount.com
`PrivateKeyID`	The private key ID can be found in the JSON key	513bd217d88b38f8f6a5fdca1fadd79bb62826e0
`PrivateKey`	This private key can be found in the JSON key of the service account. the private key must have new lines instead of \n characters.	-----BEGIN PRIVATE KEY----- MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQDBBBnTBdCnzWyO … ----END PRIVATE KEY-----
`IdentityDomain`	This is the region where we want to store buckets and objects. Available locations are listed here: https://cloud.google.com/storage/docs/locations	us-east1

XML Element Under DeviceDisk	Description	Example
`CompartmentID`	Google cloud project ID. This is is the JSON KEY too.	gcs-integration-to-diva
`ContainerName`	Target bucket name. bucket limitations are listed here: https://cloud.google.com/storage/docs/naming	
`StorageClass`	Specifies the target storage tier associated with the buckets created by DIVA. GCS support multiple storage tiers as described on this page: https://cloud.google.com/storage/docs/storage-classes	standard
`ThreadsPerTransfer`	Number of concurrent threads per transfer. Same as before.	5

Configuring DIVA for Google Cloud Storage

Configuration of a Google Cloud Storage Account is similar to the configuration of other storage accounts in DIVA. However, you must specify Client Email, Private Key Id, and Project Id fields with a Google Cloud Storage account. A Private Key and ID must be generated in the aforementioned manner. Set the vendor to Google. As usual, the creation of a Google Cloud Storage Account triggers the creation of the associated Disk Array, Disk and Actor-Disk connections.

1. In the *Edit Object Storage Accounts Entry* dialog box, select *GCS* from the Type drop-down menu. Click *OK*.
2. In the *Edit Arrays Entry* dialog box, select *ARCHIVE* from the *Storage Class* drop-down menu. Click *OK*.

   At the Array level, you can change the default storage class from *ARCHIVE* to one of the GCS-supported storage classes, such as Standard, Nearline and Coldline.

Microsoft Azure Blob Storage

Azure Blob Storage is the object storage service offered by Microsoft to Azure account owners. Like any other object storage service, Azure Blog Storage offers an interface to create blobs (objects) under buckets.

You can configure Microsoft Azure Blob Storage as you do other Cloud accounts.

See *Adding a Cloud Account* and *Creating a Cloud Bucket*.

Oracle Cloud Infrastructure (OCI)

DIVA provides support for storing your data in Oracle Cloud Infrastructure. The DIVA Web App is enhanced to support OCI storage operations. OCI services combine cloud elasticity and utility with granular control, security, and predictability of your on-premise infrastructure. OCI delivers high performance, flexibility, availability, and is cost-effective.

Note: If you have multiple DIVA sites, connected to the same OCI / OCI Classic storage account, you must use a different Array Name per site. The Array Name is used to uniquely identify content of an array in the cloud, and therefore must be different. This constraint is not required for other cloud vendors.

File Encryption Status

With AWS S3 DIVA displays encryption status in the *Server Side Encryption Settings* dialog:

With Alibaba OSS DIVA displays encryption status in the detail view of a particular file.

DIVA Web App Support

Enable SSE for S3 buckets on the DIVA Web App *CONFIGURATION* view. DIVA displays *Enable Server Side Encryption* only for S3 Cloud Buckets. If enabled, DIVA displays Server Side Encryption, KMS Key ID, and Bucket Key Enabled.

Amazon S3 Transfers

Archiving to an S3 Disk

DIVA enables you to archive content from and to a regular source such as an FTP server to and from an Amazon S3 disk. To archive an object, do the following:

1. Browse to *Content Management* > *Archive Submission*.

   DIVA opens the *ARCHIVE OBJECT* view.
2. Select a source server from the *Source Server* drop-down menu.
3. Select a quality of service from the *Quality of Service* drop-down menu.
4. Enter a value into the *Priority Value* field.
5. To execute the job, click the *Submit* button.

   Files in a folder of an FTP server are split into 5 AXF segments (determined by number of threads per transfer specified in the Storage Accounts configuration) and transferred to an Amazon S3 bucket using five threads within the actor. DIVA automatically creates the bucket named `diva-<unique bucket id>-<region>-<index>` where the index increases every 100,000 instances. This is specified in the Max Instances Per Bucket setting of the array's configuration.

   DIVA creates the unique bucket id on creation of the storage account.

Restoring from an S3 Disk

Users can restore an object on an Amazon S3 disk to a non-S3 destination, for example an FTP server. Users can also restore an object on an Amazon S3 disk as an AXF file to an S3 destination.

To restore an object from an S3 disk, do the following:

1. In the DIVA Web App, browse to *Content Management* > *Catalog Browsing*.
2. Click the three dots next to the object to be restored.
3. From the context menu select the type of restoration desired.
4. Fill in the form.
5. To execute the job, click *Submit*.

Note: Users can copy an object from one S3 disk to another S3 disk in the same AWS account, or in another AWS account. It is also possible to copy from an OCI Disk in an Oracle object storage account to an Amazon S3 disk in an Amazon account.

Oracle Storage Cloud Transfers

The Oracle Storage Cloud is an object storage solution that offers two types of accounts usable with DIVA: metered and non-metered accounts.

The non-metered account enables the creation of standard class containers. Virtual objects written inside standard containers are accessible immediately at any time.

With a metered account, DIVA can archive to standard and also to archive class containers. With archive containers, objects written to a deep archive storage device require a restoration process before they can be downloaded.

An object located in deep archive requires a maximum of 4 hours to restore to a configured target server because the content is first transferred from tape to Cloud cache, and then transferred from cache to the final destination.

When a Restore job is created for an object with a cloud instance, DIVA always attempts to restore a local (non-cloud) instance of a virtual object. If all local instances are offline, no local instances exist, or when a cloud instance is explicitly requested (a Restore Instance job), then DIVA restores from a cloud instance.

Only actors configured for CLOUD ARCHIVE can transfer content to the cloud. Only actors configured for CLOUD RESTORE can transfer content from the cloud.

For information on Oracle Storage Cloud storage accounts, see http://docs.oracle.com/cd/E60880_01/VLPFN/whatis.htm#BABDADAE

Object Storage Destinations

DIVA enables restoring content to a destination, and archiving content from a source, linked to an Oracle object storage account. Any type of object can be restored to these destinations. However, these destinations do not support symbolic links.

The Files Path Root for the destination must contain a value, and can contain an optional prefix. The value identifies the name for the target container. Use the optional prefix if the object is not to be restored to the container root directory. The prefix must be separated from the container name using either / or \. For example, `container`, `container\folder`, and `container/subdir1/subdir2` are all valid paths.

EMC ECS Object Store Integration

DIVA supports local arrays that include disks with Swift interfaces—for example, an EMC ECS object store.

During an upgrade from an earlier DIVA version, all disk instances with an `ARCHIVE` or `STANDARD` Storage Class are updated with a storage option containing both of the following:

- `-storage_location=CLOUD`
- `-oracle_storage_class={ARCHIVE|STANDARD}`

All disk instances with a `NONE` Storage Class are updated with a storage option containing both of the following:

- `-storage_location=LOCAL`
- `-oracle_storage_class=NONE`

All actor-disk connections with cloud as the interface are updated to Swift for the interface.

For detailed configuration information, see the *DIVA Installation and Administration Guide*.

Content Auto-Indexing

This chapter provides information about configuring auto-indexing.

Topics

- Supported Storage Type and Functionality
- Object Storage Auto-Indexing
- Disk Auto-Indexing

Supported Storage Type and Functionality

DIVA supports the following disks and functions:

- Object Storage auto-indexing using Cloud Buckets, S3, Azure and Google Cloud.
- Disk auto-indexing.
- The service can index a file from any type of storage class.
- DIVA creates an AXF_RF_1.1 reference-file metadata for each file detected.
- One file per object in this release.
- Empty files are excluded.
- Overwriting existing file is prohibited.
- Direct file deletions aren't supported; deletes must be sent to a DIVA API.
- A trial mode is available to be able to test the behavior of auto-indexing.
- Azure blob tags are only supported by v2 storage accounts for general purposes; object auto-indexing doesn't work with v1 storage accounts.

Object Storage Auto-Indexing

DIVA supports Object Auto-Indexing as well as Cloud Bucket Scanning. You can trigger both types of scan from the context menu for the relevant disk, either from the *CLOUD DISKS* view, or from the *PROPERTY* view of a specific disk.

Content Indexing is the name for the Object Auto-Indexing scan. This option is available only if the Cloud Bucket format for the disk is set to `AXF RF 1.1`.

When you select Start Cloud Bucket Scanning, DIVA displays a Cloud Bucket Scanning form.

Selecting *Content Indexing* as the scan type displays the dialog for the options pertinent to that type of scan.

Auto-indexing offers some options when the service is started, as follows:

Option	Description
Index files created after [date/time]	The service considers only the index files created after the specified date and time.
Default Category	Depending on the value of Object Path Prefix, the service can use a default collection for all the objects. If the scan requires a default collection, set this parameter. Otherwise, DIVA doesn't require a Default Category setting.
SubPathFilter	If the auto-indexer should detect only the file path starting from SubPathFilter, set values for the SubPathFilter. See *Cloud Array Parameters for Object Auto-Indexing*. Example with these parameters set: • The configuration of Object Path Prefix • Bucket name is set to `MyBucket/MyVideoFiles` • SubPathFilter is set to `ABC` In this example, auto-indexing considers only the cloud objects starting from `MyVideoFiles/ABC` inside `MyBucket`.
Trial Mode	You can specify Trial Mode when starting a scan. This mode of operation doesn't save objects to the database and doesn't update the objects in the cloud. Since changes to AXF metadata files and database entries are difficult to revert, Telestream recommends testing settings before running auto-archiving. You can use Trial Mode for testing some settings before auto-indexing. **Note:** Because there is no persistence in Trial Mode, duplicated objects (same name, same collection) aren't detected. Auto-indexing attempts to detect files created or modified since a specific date and time. The date keeps updating and auto-indexing will attempt to detect new files created or modified 12 hours before this date and time. The overlap between each listing iteration makes it possible to see the same object multiple times in Trial Mode.

Scan events appear in the *Scan Events* table on the *Property* view of the cloud disk.

You can't run both types of scan against a bucket. The default view of the properties table shows all scan events. However, you can filter the *Scan Events* table by scan type.

Cloud Array Parameters for Object Auto-Indexing

The behavior of the service also depends on the configuration of the cloud disk associated with the service.

- **Format**—The auto-indexing feature generate AXF_RF_1.1 instances. It works only when the format of the cloud bucket is set to AXF_RF_1.1.
- **Bucket Name**—DIVA scans and indexes files under `<BucketName>[/Directory]`. The bucket name can include an optional subdirectory; if it does, DIVA scans only the contents of this subdirectory.
- **Object Path Prefix**—The behavior of the auto-indexer depends on the selected prefix.

This table describes how auto-indexing sets Objectname, Collection, and UUID depending on the Object Path Prefix selected:

Object Path Prefix	Objectname	Collection (Category)	UUID	Comment
None	Name of the file without the extension.	Default collection that is set in the configuration of the service.	Generated by the actor.	
Name&Category Auto-indexing will only process the files starting from two sub-folders after `BucketName[/directory]`.	subfolder2	subfolder1	Generated by the actor.	ObjectName and Collection are extracted from the path using a regular expression containing three patterns in parenthesis: `([^/])/([^/])/?(.*)` • The first is the collection. • The second is the Objectname. • The third is the relative path of the file within the DIVA object. The file paths not matching the regular expression are ignored.
UUID Auto-indexing will only process the files starting from a sub-folder matching a UUID.	Name of the file without the extension.	Default collection that is set in the configuration of the service.	subfolder1	The UUID is extracted from the path using a regular expression: `([0-9a-f]+-[0-9a-f]+-[0-9a-f]+-[0-9a-f]+-[0-9a-f]+)/?(.*)` The file paths not matching the regular expression are ignored.

Auto-Index and AXF Archive Locations

The auto-indexing service can generate AXF_RF objects and populate the database with new objects.

Telestream recommends starting the auto-indexing service on a cloud bucket not already in use. Starting the service on a cloud bucket that has already been used by DIVA to store AXF instances could be disruptive. For example, if auto-indexing is started with a cloud array that already contains AXF_RF objects, the auto-indexing may create additional AXF_RF objects pointing to the same content. If one of the objects or

instances is deleted, DIVA deletes the content of the second object or instance also. The same could happen if you archive or copy to a cloud array that is already auto-indexing.

To prevent that situation, DIVA creates a small file on the cloud array describing the use of the cloud array (AXF-ARCHIVING, AXF_RF-ARCHIVING, AUTO-INDEXING). The name of this file is the name of the array with a USAGE extension. If DIVA tries to use the same array for different purposes, the operation fails.

If you start the auto-indexing service, but the cloud array has already been used for archives or copies, the actor terminates auto-indexing with the following error in the event log:

```
Can’t use this disk location for AUTO-INDEXING purpose because
it's already in use for AXF_RF-ARCHIVING.
```

If you try to archive or copy to a cloud array that has already been used for auto-indexing, the job aborts with the following error message:

```
Can’t use this disk location for AXF_RF-ARCHIVING purpose
because it's already in use for AUTO-INDEXING.
```

See also *AXF Discovery*.

Clearing Files from Failed Auto-Indexing

If an object already exists in DIVA with a UUID different than the one discovered during the auto-indexing of a cloud disk, DIVA skips the object and posts an ERROR event. This could leave the file associated with an AXF metadata file that isn’t referenced in the database; a kind of an orphaned AXF instance.

You can retrieve the event from the API using *GET /disks/scans/*events.

From the data bucket, the file that was not indexed has a tag containing the UUID of the object that auto-indexing tried to create.

To clear out duplicated files, do the following:

1. Navigate to the AXF metadata location and delete the associated `<uuid>.axf` file. (for example, 82aa75b1-fb55-ed11-b6bf-1b65982d7a18.axf).
2. Upload the same file under a different name so that auto-indexing will generate a different objectname or different collection.
3. Delete the file that the auto-indexing failed on.

Disk Auto-Indexing

You can synchronize the content of a directory with DIVA. DIVA monitors the content of the directory as a watch folder, detects a specific list of file-system operations, and indexes the directory as these AXF object instances:

- New files
- Modified files
- Deleted files

- Renamed files/folders

DIVA automatically associates each file of the watch folder to an object. You can use the object in DIVA workflows such as:

- Copies, Restores, and Partial Restores
- Duplication via storage policies
- Proxy creation
- Metadata extraction

Supported Functionalities

- Disk auto-indexing can monitor a directory from LOCAL, DISK or CIFS unmanaged server (aka source/destination). Other types of unmanaged servers are not supported.
- Disk auto-indexing can store AXF metadata to a disk connected to an actor via local or remote (SMB) connection.
- A disk array may contain one or multiple disks. For the purpose of disk auto-indexing, the recommendation is to create a dedicated array with a single disk.
- A disk used by auto-indexing to store AXF metadata cannot be used for regular archives. The reverse is also true.
- Long path names are supported
- Disk auto-indexing can monitor a network share if the protocol is SMB 3.0. Older SMB versions are not supported.

Configuration

To be able to use disk auto-indexing, a LOCAL, DISK or CIFS unmanaged server and a local or remote managed disk must be configured.

A dedicated disk array containing a single disk shall be configured. The actor-disk connection must be local or remote.

At least one actor must be able to access the disk and the unmanaged server.

Copy and Restore Request

In the case of a restore request, the selected actor must have access to the:

- managed disk
- unmanaged server that was used by auto-indexing
- target unmanaged server for the restore

In the case of a copy request, the actor must have access to the:

- managed disk
- unmanaged server that was used by auto-indexing

Starting and Stopping the Scanning Service

To start or stop the auto-indexing service, browse to *Resource Management* > *Unmanaged Storage*.

At the start of a scan, all actor-disk connections are initialized to READ-ONLY / READ-DELETE depending on whether instance deletion is enabled or not.

The following table contains descriptions of the auto-indexing parameters.

Parameter	Description	Default Value
Allow duplicate objects	If DIVA needs to create a new object but there is already another one under the same name and collection, DIVA can automatically add a suffix (number between parenthesis). When this option is set to No, DIVA will ignore and skip the new object found during auto-indexing. For more information, the “Duplicates handling“ section below.	Yes
Disk	Name of the disk that is used to store the AXF metadata of all the indexed objects.	No default, must contain a valid disk with a local or remote connection to an actor.
Watch folder root	Absolute root directory to monitor on the unmanaged server. If empty, DIVA will use the files path root configured for the unmanaged server.	Empty
Filters	A filter is a string based on wildcards or a regular expression associated with metadata mappings. Multiple filter can be defined. They are evaluated one by one to see if a file must be indexed or not. Only one filter can be triggered for a given new file found on the watch folder.	At least one filter must be configured.

The following table describes all the parameters for a given filter:

Parameter	Description	Default Value
Is regular expression?	Boolean switch indicating whether the subsequent filter expression must be interpreted as a regular expression or a wildcard expression.	True
Filer expression	Regular expression or wildcard string that will be used by DIVA to check if a given file must be indexed or not (the relative path of each new detected file is evaluated). If this field is empty, all the files will be indexed by the filter.	Empty
Objectname	This string will be used by DIVA to build the object name of each new file detected. This parameter can only be used when the filter expression is a regular expression. A regular expression can be composed by multiple sub-expression isolated between parenthesis. The string matched within the parentheses can be reused to build the object name. Don't enter a static string in this parameter, otherwise all the files detected will have the same objectname. If this parameter is empty, DIVA will use the name of each file as objectname.	Empty
Category	Like for the objectname, this parameter may contain the string matched by a sub-expression. This parameter must contains something, even a static string of all the files matching this filter must be in the same category (collection).	Can't be empty
Comment	This is a string of characters containing the comments for all the objects created by this filter.	No default value
Storage plan	Name of a valid storage plan. All the files indexed by auto-indexing and then objects created by this filter will be associated with the storage policy specified by this parameter.	Can be empty if DIVA should not associate the new objects to a storage policy.

Filter with Regular Expressions

Wildcard-based filters are simple and intuitive, but limited. Filters based on a regular expressions offer more advanced capabilities like, such as:

- Generate values for objectname, category or comments based on substrings found in the path of each file
- String extraction based on sub-expression.

- More specific pattern matching

File-system Action Handling

There are four kinds of filesystem action that are captured by auto-indexing: files added, modified, deleted, renamed. When one of these actions are detected, it is handled by its own thread. Auto-indexing can address multiple actions in parallel. This chapter describes how these actions are handled by auto-indexing.

On File Added

When a new file is detected by auto-indexing, the first step is to wait until the file is no longer locked by another application. This may happen if the file is still growing of if there are still modifications in progress. As soon as the file is no longer locked, the auto-indexing thread for this file will get the file size and compute an MD5 checksum. This checksum is used for future data integrity check and to verify is the file has not already been archived before (see the next section about duplicates handling). This action could end with a new object (normal behavior), a new instance of an existing object or the new object is ignore because duplicates are not allowed by configuration.

On File Modified

The detection of modified files is performed by comparing the last modified time of a file with the creation time of the AXF instance. When a file has been modified, DIVA will deleted the existing instance (potentially the object if it's the only instance) and re-index it. When a file is re-indexed, its relative path is evaluated against the current filters. If no filters are matching, the file will be ignored.

On File Deleted

When the auto-indexing service detects that a file has been deleted on the watch folder, DIVA will delete the associated object disk instance. If this is the only instance for a given object, the object is deleted from the database. The associated AXF metadata are deleted after the database deletion.

On File Renamed

Auto-indexing can detect when a file has been renamed. When it happens, the instance associated with the old filename is deleted and a new object in created according to the filter in place at this time.

On Sub-directory Renamed

When a sub-directory of the watch folder is renamed, the action is detected on service startup and in real-time too. All the files that were indexed under a renamed folder are

first removed from the database and re-indexed as new objects/instances. Depending on the number of indexed files under a renamed subfolder, this action could take quite a lot of time.

Duplicates Handling

Object names, collections and UUIDs are generated by the auto-indexing service regardless what is already recorded in the database. Depending on the content found on the watch folder and the configuration of the filters, auto-indexing may generate objectname/collection pairs that already exist in the database. There are two types of duplicates:

True duplicates. Same objectname, same category, same file and same checksum. DIVA generates a checksum for each file during auto-indexing in order to verify that the indexed file is the same as the one already in the archive. When a true duplicate is detected, DIVA adds an instance to the existing objects.

Objectname/Collection duplicates. These duplicates are generated based on the filters and the content indexed. The difference with true duplicates is the content of the archive. It's not the same file because its checksum is different. When this kind of duplicate is detected, DIVA adds a suffix to the objectname so that the new indexed object is unique.

Instance Delete Behavior

Auto-indexing detects and handles file deletions automatically. However, if an object containing an instance created by auto-indexing is deleted via API or by the DIVA Web App, DIVA takes some actions on the watch folder to prevent the file from being indexed later. DIVA can't delete them because they are on an unmanaged server. Instead, DIVA moves the deleted files under a special directory named `.DivaRecycleBin`.

Note: `.DivaRecycleBin` is created on the first delete request under the watch folder, but its content is not captured by auto-indexing.

AXF Discovery

This chapter describes AXF Discovery.

Topics

- About AXF Discovery
- Supported Functionalities
- Scanning Service Start and Stop
- Scan Events
- Scan Statistics
- Rescan of Existing Objects and Instances
- Rescan of Bad or Incomplete Instances
- Automatic Scan Restart
- Actor-Disk Connection Access Type

About AXF Discovery

DIVA supports the discovery of non-complex objects and instance metadata from an object storage account to a Core database. This allows you to update an on-premise DIVA system with content from the cloud.

This feature keeps scanning a cloud bucket containing AXF instances (AXF and AXF_RF). This bucket is populated by a third-party software:

- Can be a bucket replication software
- Can be another DIVA system

In both cases, DIVA scans the bucket for new objects and populates its database when new objects are detected.

See also *Auto-Index and AXF Archive Locations*.

Supported Functionalities

DIVA supports the following features:

- Managed disks using S3, Azure, Google Cloud and OCI are supported. Other types of disks aren't supported (for example: Local, or NAS).

- AXF_1.1 self-contained and AXF_RF_1.1 reference-file are supported and the format is automatically detected by the scan.
- DIVA ignores bad or incomplete AXF instances and doesn't rescan them.
- DIVA doesn't support complex objects. The service supports small objects (1-50 files per object).

Scanning Service Start and Stop

The scanning service is not a separate component or a Windows service; it is a functionality of DIVA. Part of this service is executed by the Manager and part is executed by an actor selected automatically by Manager.

The service associated with the cloud disk to scan is started from the DIVA REST API. There is an endpoint to start the scanning service.

The service part running on the actor ends only when the actor is stopped if requested by the Manager. Otherwise, it keeps scanning the bucket for new AXF instances.

Scan Events

During a scan DIVA persists events detailing the progress of the scan. The events can be retrieved using the DIVA REST API by calling *GET /disks/scans/events*. Events may be paged and filtered by the following fields:

- scanId
- serviceId
- diskName
- actorName
- objectName
- category
- description
- severity (pull-down menu)
- type (for example: AXF_SCAN, pull-down menu)
- eventType (for example: SCAN_STATISTICS, pull-down menu)
- event date-time range
- last scan date-time range

Note: Data that is null is not available for that event type. All times are epoch time in seconds.

Scan Statistics

When the scan service goes through all the objects of the list, the actor sends statistics to the Manager containing:

- The number of instances successfully detected.
- The number of bad or incomplete instances.
- The epoch time when the list started (indicates the creation/modification time of the last instance detected).
- The duration of the list scan.

To retrieve the scan status of a disk, use the `/ui/disks` endpoint. The status can be one of three values:

- NONE
- RUNNING
- STOPPED

Rescan of Existing Objects and Instances

When the service is stopped and restarted to scan from a specific date/time, the actor may detect instances that have been previously entered in the database. This case is covered by the Manager. The instances found by the actor is ignored by the Manager, and there is no overwrite of the database entries.

Rescan of Bad or Incomplete Instances

A bucket may contain bad or incomplete instances for various reasons:

- An instance is corrupted or incomplete because of a previous aborted job.
- An instance is incomplete because it has not been fully replicated yet.

The scanning service running on an actor keeps track of the instances that failed validation during a scan.

The actor stops testing them again after 100 retries. If there is a need to force a rescan of a bad instance beyond 100 attempts, stop the scan service first, then download, edit and overwrite a new version of the `BadInstances.json` file. The number of retries can be changed manually before restarting the service. The following is a sample `BadInstances.json` file:

```
{
"BadInstances": [
{
"axf-file": "13ba7b81-9dfd-12d1-80b4-10c040243460.axf",
"retries": 50
```

```
},
{
"axf-file": "e3f4da0e-5bce-4314-817d-bf268c796c5c.axf",
"retries": 4
}
]
}
```

Automatic Scan Restart

If the Manager or Actor service crashes during a scan, the scan restarts with the start of the corresponding service. It can take up to ten seconds following the start of the service for the scan to automatically restart.

Actor-Disk Connection Access Type

DIVA supports true READ-ONLY, and READ/WRITE/DELETE connections.

Note: If you change a connection's access type to Read/Write/Delete while a disk is being scanned, DIVA doesn't allow write access until the scan is stopped.

The *CLOUD DISKS* view in Resource Management for DIVA displays the cloud disks and their scanning status in the main table:

Start Cloud Bucket Scanning from the Action menu for an individual disk. When a Cloud Disk is running a scan, the Action menu also allows you to stop the scan.

DIVA displays the *Cloud Bucket Scanning* dialog. Enter the scan settings.

The *CLOUD DISK PROPERTIES* view displays the scan events associated with the disk in a table at the bottom of the view. Events can be filtered by Severity, Type, actor, Object, or Description. You can also control the scan state from the property view in the Actions drop-down menu.

Data Management on Removable Media

Removable media refers to tape, ODA cartridge or ALTO disk storage. This chapter contains information specific to configuration and operational aspects of removable media.

Topics

- Tape Groups and Sets
- Robot Manager Configuration
- Tape Group Configuration
- Tape Drive Reservations
- Tape Label Management
- Tape Drive Encryption
- Tape Compression
- Tape Spanning
- True Remaining Tape Size and Last Written Position
- Repacking Tape Jobs
- Verifying Tape Jobs
- Tape Eject/Insert
- Export and Import Functions
- Adding a Tape Library, Corresponding Drives and Tapes into DIVA
- Damaged Tape Utility Recovery
- Tape Scanning for Indexing Objects

Tape Groups and Sets

Disks are logically assigned to arrays for the storage of objects, but tapes are logically associated together in tape groups.

Tapes are initially divided into sets and assigned a number referred to as a *set ID*. A set ID enables partitioning of pools of tapes in a library and assigning them for use with

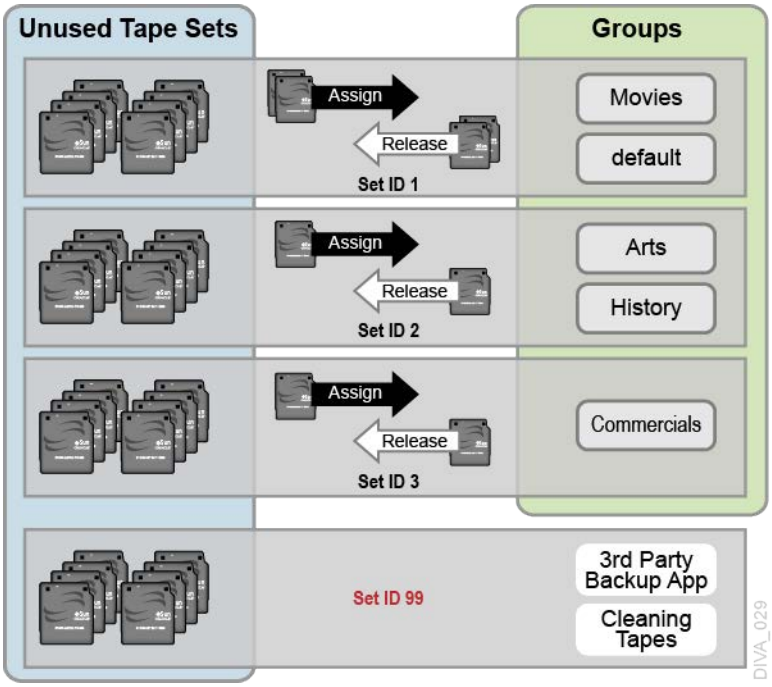
specific tape groups. A tape group draws upon the pools by associating the tape group with a set ID.

More than one tape group may use the same Set ID. An unused tape doesn't actually belong to any of those tape groups until DIVA writes the first object to that tape. When all objects have been deleted from a tape group-assigned tape, it is unassigned from that tape group, and may subsequently be assigned to another tape group using the same Set ID.

Since tape groups are a user-defined concept they can differ from one DIVA system to another. The exception is the Default Tape Group, which exists in all installations and can neither be renamed nor removed.

When a tape is assigned a Set ID of 99 it indicates to DIVA that the tape is not to be used and is not related to the operation of DIVA. Examples are tapes that belong to a non-DIVA application in a shared library environment or the library's cleaning tapes.

This figure illustrates how tape sets and tape groups are associated and used:



Viewing Tape Groups

In a DIVA system, tape groups are created and managed with the DIVA Web App on the *TAPE GROUPS* view.

To view the tape groups, browse to *Configuration* > *Library Storage* > *Tape Groups*.

Robot Manager Configuration

Although you can use DIVA to manage disk storage, you can expand storage capacity by adding one or more tape libraries, also known as medium changers. In these cases, the Robot Manager module provides an intermediate software layer for DIVA to interact with many different types of medium changer. It is connected to DIVA through TCP/IP.

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

The Robot Manager alerts DIVA when the collection of tapes in the associated library requires synchronization with the Core database. This functionality is specific to the SCSI Robot module, and detects potential tape inventory mismatches between Manager database and the library inventory.

Potential inventory issues are trapped by the Robot Manager if the library sends unit attention codes 06h 00h 28h (inventory may be altered), or 06h 00h 29h (reset occurred). When this happens, the Robot Manager notifies DIVA so that it resynchronizes the database with the content of the library.

The Robot Manager Client is accessed using either the Robot Manager Client (command line based) or the Robot Manager Client GUI. The Robot Manager Client GUI is a graphical interface making it easy to interact with the Robot Manager.

Note: If intermediate robotics control software is installed between the Robot Manager and the library under control (such as ACSLS, SDLC, or PSC), it must be running before starting the associated Robot Manager.

When the Robot Manager Client command-line interface is started it displays a page similar to a Windows command line. The Robot Manager Client is already started and only the commands necessary to perform the required operations, or to display the required information, need to be entered.

The following list are the Robot Manager executable files:

Caution: Although a Robot Manager may be restarted while DIVA is running, an attempt to mount a tape to a drive while the Robot Manager is offline may cause the drives to be set to Offline.

- `%DIVA_HOME%\Program\RobotManager\bin\RobotManager.exe command [options]`

  Executes commands for the Robot Manager Service. Appending the `-conf` (or `-f`) option after one of the following commands specifies a specific configuration file to load settings from. The RobotManager.exe command parameters are as follows:

  – `install` (`-i`)

    Installs the Robot Manager Service as set by the SERVICE_NAME parameter defined in robotmanager.conf. If this parameter is undefined, the service is installed as Core Robot Manager—host_name.

  – `uninstall` (`-u`)

    Removes the Robot Manager service.

  – `debug` (`-d`)

    Starts the Robot Manager in console mode.

  – `version` (`-v`)

    Displays the Robot Manager version information and then exits.

  – `help` (`-h`)

    Displays help information and then exits.

- `%DIVA_HOME%\Program\RobotManager\bin\RobotManagerClient.bat [rmHost] [rmPort]`

  This is a command line utility to take control of the Robot Manager if the DIVA system is down.

  – `rmHost`

    The remote host name for the connection.

  – `rmPort`

    The remote host port for the connection.

- `%DIVA_HOME%\Program\RobotManager\bin\RobotManagerGUI.bat`

  This is a GUI utility to take control of the Robot Manager if the DIVA system is down.

Configuration Overview

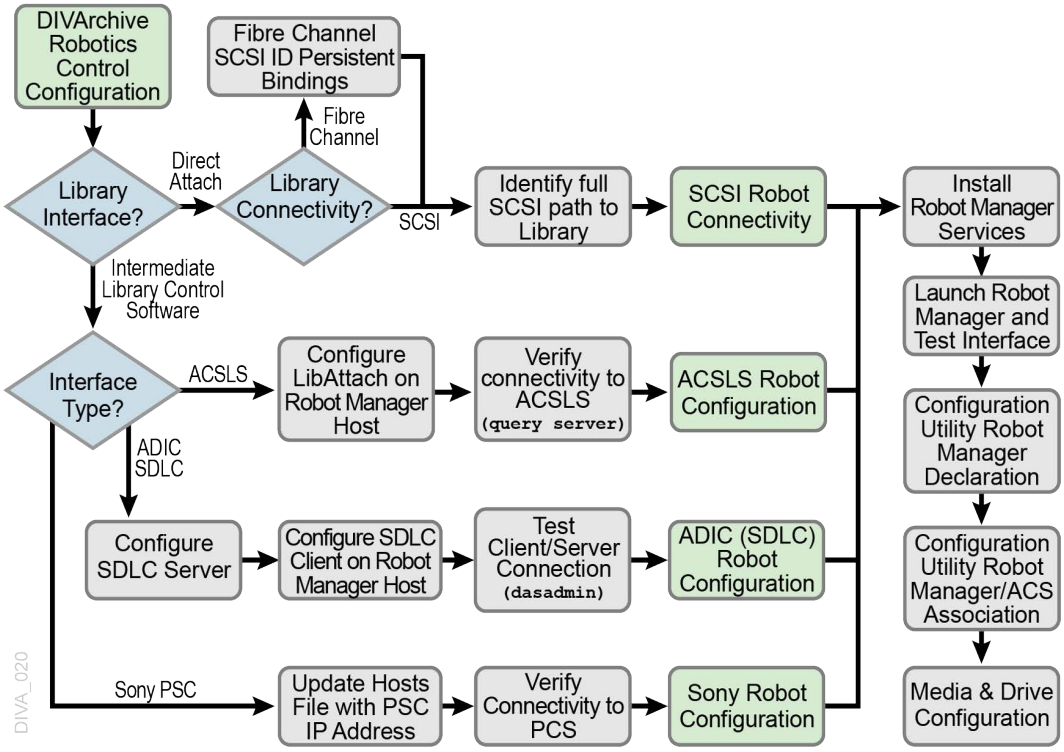
The Core Robot Manager on Windows platforms runs as a Windows service and is launched automatically with Windows.

These are the main configuration steps for the Core Robot Manager:

You configure the type of interface a specific library in a static configuration file. The file name is robotmanager.conf and is located in the `%DIVA_HOME%\Program\conf\robot_manager` folder on the computer where the Core Robot Manager is installed. In a new installation (or upgrade) the file is provided with an INI extension. You must copy the file, remove the INI extension, and then edit the new file.

Since many different types of Managed Storage and connections are supported, not all sections of the configuration file will be relevant to your particular installation. Also, some parameters are specific to the operating system where the Robot Manager is installed. Therefore, some settings in the configuration file are initially commented out (that is, they have # in front of the parameter). This indicates to the Robot Manager to ignore the setting. For the setting to be taken into account the # must be removed.

The following figure outlines the steps for configuring the robotics to be controlled by DIVA:



Common Robot Manager Parameters

The following table identifies common parameters that are applicable to all the medium changers.

Parameter	Parameter Type	Description	Default
SERVICE_NAME	Name	The display name of the Robot Manager Windows Service. You must set this variable if multiple Robot Managers are installed on the same server. If this variable is used, the Service Name is *DivaRbt-<SERVICE_NAME>*. If this variable is not set, the default Service Name is *DivaRbt*.	Uncommented
RM_PORT	TCP port number	The TCP port that the Core Robot Manager listens on for incoming jobs. This value must be unique if there are multiple Core Robot Managers running on a single host computer. This is typically, TCP port 8500 and greater.	8500
RM_ACS	Number	The ACS (Automated Cartridge System) controlled by the Core Robot Manager module. This value will appear in the Robot Manager/ACS Association List in the DIVA Web App for this Robot Manager after database synchronization	0

Direct Attached SCSI Medium Changer Settings

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

In either case, the Core Robot Manager uses its own DIVA-provided driver (SCSI_Robot.dll) to directly interface with the library without the need for intermediate library management software. For this type of SCSI-attached library, you must uncomment the entries in the following topics, and configure them in the `robotmanager.conf` file. Library Drive Models and Tape Types parameters are located in other configuration files.

Common Settings for SCSI-based Medium Changer

The following are typical settings for the SCSI-based Medium Changer:

Robot Manager Common Options

Uncomment only the RM_MODULE=SCSI_Robot.dll in the Windows environment.

SCSI Module Parameters

The following table identifies common SCSI module parameters.

Module Parameter	Operating System	Description	Values
RM_SCSI_DEVICE_LSM0	Windows	This specifies the SCSI target of the library as it identified by the host operating system.	ScsiP:B:T:L
RM_SCSI_DEVICE_LSM1	Windows	This setting is specific to a StorageTek dual L1400M library with a PTP (Pass Through Port), and specifies the SCSI target of the 2nd frame (LSM). This parameter can also be set to the serial number of the medium changer. Although this type of library configuration can be addressed using only the RM_SCSI_DEVICE_LSM0 connection, DIVA manages this type of library more effectively when both frames are specified. DIVA also manages the PTP in this case.	ScsiP:B:T:L

The SCSI Library Connection

For the SCSI interface Managed Storage the Core Robot Manager communicates with the library directly over the SCSI hardware layer and doesn't require the Windows driver interface.

For all other Managed Storage it is essential that no library driver be loaded for the library interface. If a driver is loaded, the Core Robot Manager will be unable to communicate with the library. In this case, if your library doesn't appear in Windows Device Manager as an Unknown Medium Changer, the Robot Manager will be unable to communicate correctly with the robotics.

If you can't 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.

For Windows, you can determine the RM_SCSI_DEVICE_LSM(n) settings for the Core Robot Manager using the scandrive.exe utility. The utility is located in the `%DIVA_HOME%\Actor\bin` directory. The utility automatically reports all devices located in the Windows SCSI hardware tree in the registry and their corresponding Port, Bus, Target, and LUN (Logical Unit Numbers).

The utility reports the SCSI Device ID of the library in the format ScsiP:B:T:L (see the previous figure), where P is the port number, B is the bus number, T is the target number, and L is the Logical Unit Number.

The Type section of the utility's output refers to that peripheral's class (HDD, CDROM). A tape library is reported as a Medium Changer Peripheral, and the Identifier for each corresponding device reported should match the model number of the library itself (for example, SL500). You can then enter the full SCSI path reported for each library into the RM_SCSI_DEVICE_LSM(n) settings in the robotmanager.conf file.

Additional Settings

The following table identifies an additional parameter that can be set to change the behavior of Robot Manager with a SCSI Medium Changer.

Parameter	Parameter Type	Description
RM_SCSI_ENABLE_MEDIA_TYPE_DETECTION_LAYOUT	String pattern	The purpose of this parameter is to detect the type of a media from the volume tag returned by the library. The layout is a string of 8-10 characters indicating where the label and the mediatype are. It must contain these three characters only: L: The character at this position is part of the tape label/barcode considered into DIVA Database. T: The character at this position will be used for media type detection X: The character at this position will be ignored Example: for a given volume tag ABC003L6, if the layout is set to LLLLLLTT, RobotManager will detect an LTO6 tape and report ABC003 to DIVA.
RM_SCSI_MOVEMEDIUM_TIMEOUT		Robot SCSI uses the MOVE MEDIUM SCSI command during mount, dismount, enter, and eject jobs. The value of RM_SCSI_MOVEMEDIUM_TIMEOUT is indicated in minutes, and the default timeout is fifteen minutes for the communication between the library and the robot manager. Some Managed Storage, like Spectra T950, may require more time to be able to complete a MOVE MEDIUM job and you should set this parameter value accordingly.
RM_SCSI_EJECT_USEGLOBALLOCK		You must set this parameter to one if you want the SCSI Robot Manager Eject calls obtain the lock number of the LSM and hold that lock until all associated tapes to be ejected have completed the ejection process. When all tape ejections are complete, the call unlocks the drive and proceeds on to the next drive. The default setting is zero.

Fiber Channel HBA (Host Bus Adapter) and SCSI Persistent Binding

Most installations use Fiber Channel (FC) rather than native SCSI to interface to the library (typically over a SAN). In these instances, the FC HBA in the Core Robot Manager host presents the World Wide Name of the library interface as a SCSI ID. By default, most

HBAs automatically map these to a SCSI ID for the host operating system to access. This presents a problem if a device is added or removed on the SAN because it could alter the SCSI ID of the library by the HBA, and automatically remap the existing devices. Disable the Automap feature to avoid this issue and use Persistent Bindings instead. This feature allows the SCSI mapping of the library to remain consistent between host restarts, and from the advent of any addition or removal of devices on the SAN.

If the library controller or the HBA in the Core Robot Manager host is changed, this might alter the library's SCSI Persistent Bindings to the host operating system. This requires the Persistent Binding for the library to be reconfigured in the HBA configuration software on the Core Robot Manager computer.

Sony ODA Drives

DIVA supports the Sony optical drives and their WORM media (using a UDS format). Only AXF formatted objects can be written to Blu-ray discs. The drives are controlled by the Robot DIVA and the media is viewed as a tape cartridge.

These drives are shown as Unknown Medium Changer under the Medium Changer devices section in the Windows Device Manager because there are no device drivers for them. The drive itself also displays as an Optical SCSI device with the make and model number under the Disk Drives section.

There are different types of disc media available for use with Sony optical drives:

Media type	Capacity in KBytes
SONY-ODC300R	293,265,408
SONY-ODC300RE	293,265,408
SONY-ODC600R	586,530,816
SONY-ODC600RE	586,530,816
SONY-ODC1200RE	1,173,086,208
SONY-ODC1500R	1,500,020,736
SONY-ODC3300R	3,222,717,696
SONY-ODC-5500R	5,372,184,576

The disc types are identified in the `scsi_tape_types.ini` file (described in the following topic).

Note: You must configure the drive settings before configuring DIVA. The recommended parity setting is PARITY ON.

You can view the drive specifics using the Optical Disc Archive Utility. This utility enables viewing of device logs, and viewing and changing drive settings.

To change the drive settings, click the *Setting* view in the Optical Disc Archive Utility. Technical Support recommends leaving the Default Volume Type set to PARITY ON, and to use the default settings for the remaining items.

Click the Media item under the Drive navigation tree to view information about the media in a drive.

You click the Write-protect button to write-protect a drive. Once an Optical Disc is write-protected, you can no longer write objects to the device. However they are still retrievable.

Configuration File Adjustments

You must change several parameters in the `scsi_drive_types.ini` configuration file to use these optical drives.

In the `robotmanager.conf` configuration file, under the SCSI module specific options, the serial number must be identified. You can find the serial number in the `RM_SCSI_DEVICE_LSM(n)` parameter line. For example, `RM_SCSI_DEVICE_LSM(0)=00001003`, where (0) is the LSM number, and 00001003 is the serial number. You must identify the serial number for all listed devices (LSM(0), LSM(1), LSM(2)).

In the `scsi_drive_types.ini` file, the drive types must be uncommented (remove the #). For example, remove the # from in front of the line that reads #601 0x00 0x00 SONY-ODS-D77F 600 601 602 603 604 605 to use your D77F drive as shown. The TransportDomain and TransportType are obtained automatically and not used in the configuration, so you must leave these set to 0x00 as shown in this example.

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

Also, in the `scsi_tape_types.ini` file, uncomment all of the disc types listed as shown in the following example. The R or RE after the disc number indicates whether the disc is Write Once (R) or Rewritable (RE). This indicator is used because the barcode doesn't 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
```

DIVA Web App Settings and Information

You must configure the following settings in DIVA:

Drives View

Set the Drive Properties to 64 KB. The serial number comes from the Robot Manager and the firmware release number comes from the drive.

Tapes View

The Tape Properties area displays all of the enabled Tape Types from the `scsi_tape_types.ini` file.

DIVA Web App Settings and Information

The Optical Drives and Discs are displayed in DIVA on the *DRIVES* view as Tape Drives and Tapes respectively.

Repack of the discs and deletion of objects is available. However, the space is not recoverable. When trying to repack the disc, the normal Repack dialog box is displayed, but there is 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.

Additional Information

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

- Because Write-Once media must be finalized, zero remaining space will be reported to the Manager.
- Objects are spanned when there is 100 MB of space remaining. This is so that there is space left for the disc to 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 500 MB of space remaining unless an object was spanned. However you can manually finalize the disc through the Optical Disc Archive Utility.

- If a drive is manually mounted and viewed in the Windows Explorer, the display will show the individual files on the disc. Each file name will begin with a numeric value at the beginning that identifies the object's location on the tape.

ACSLS Attached Libraries

DIVA can directly interface to most Oracle StorageTek Managed Storage using the Robot_SCSI driver. Some library configurations require the use of the Oracle StorageTek ACSLS library management software for the Robot Manager to control the library.

You can install ACSLS (Automated Cartridge System Library Software) only on Solaris platforms. The Solaris host and ACSLS are sold and supported by Oracle.

For detailed information, see the Oracle ACSLS documentation at http://docs.oracle.com.

Technical Support doesn't support DIVA installations under the Solaris operating system.

LibAttach

LibAttach is an intermediate Windows driver providing connectivity to the ACSLS host. LibAttach runs as a Windows service and is typically installed on the same computer running the Core Robot Manager. The Core Robot Manager communicates to the ACSLS host using the LibAttach driver.

You must enter the following settings on the LibAttach Configurator dialog box (part of the ACSLS software):

Library server host name

Host name or IP address of the ACSLS server. If you use a host name, it must be resolvable by the Core Robot Manager host.

Firewall support

These settings are only required if a firewall is installed between the Robot Manager host and the ACSLS server. If no firewall is present leave these parameters set to 0.

The LibAttach Connectivity to ACSLS

You can verify connectivity from the Robot Manager host to the ACSLS server with the `query_server.exe` utility located in the LibAttach installation directory. When you launch the utility a Windows command prompt opens. DIVA returns statistics from the library if the connection is successful.

Firewall Support

You must have a TCP or UDP port open (to allow communication) if there is a network firewall between your Robot Manager host and ACSLS server. If there is a firewall, enter the open port numbers into the Firewall Support settings in the LibAttach Configurator.

Early implementation of firewall support for LibAttach did not work correctly with the Core Robot Manager, even though the query_server utility returned a successful connection. Ensure that you have the latest release of LibAttach that incorporates the patch released to address this issue. Contact Technical Support for additional information.

robotmanager.conf Common Options

This table identifies common `robotmanager.conf` parameters:

Parameter	Parameter Type	Description	Default
RM_MODULE=ACSLS_Robot.dll		Uncomment only this line	Commented
RM_ACS	Number	ACSLS configurations ignore this value because the ACS number is supplied from ACSLS.	Ignored

The following table identifies the ACSLS parameters:

Parameter	Parameter Type	Description	Default
RM_ACSLS_SERVER	IP Address or Host Name	ACSLS ignores—leave blank.	
RM_ACSLS_TIMEOUT	Time in milliseconds	Timeout period (ms) for queries to ACSLS via LibAttach. If 0, timeout period used by Robot Manager is 10 minutes.	0

Parameter	Parameter Type	Description	Default
RM_ACSLS_IE_TIMEOUT	Time in milliseconds	When an Insert or Eject tape command is issued you must open the CAP and insert or eject tapes within this timeout period. If you leave this value set to 0, the timeout period used by the Robot Manager is 10 minutes. If you must alter this timeout period, replace 0 with your own value (in milliseconds).	0
RM_ACSLS_MAX_DISMOUNT_RETRIES	Number of retries	The maximum number of retries when the dismounted drive is still in use. If the setting is 5, the initial delay is five seconds and then doubled after each retry.	5
RM_ACSLS_DISMOUNT_FORCE	0 (disabled) 1 (enabled)	Under normal circumstances, you must unload a tape first (using an actor) before issuing a dismount command to the library. A forced dismount instructs the library to issue the unload command to the drive directly. This option is not recommended because this may interfere with operations on the actors.	0

Sony PetaServe Library Configuration

Control of Sony PetaServe library from the Robot Manager is directed through the Sony PSC controller over an Ethernet connection. The PSC controller parameters for the Robot Manager configuration file must match those on the PetaSite Controller.

robotmanager.conf Common Options

The following table identifies common `robotmanager.conf` options:

Parameter	Parameter Type	Description	Default
RM_MODULE=SONY_Robot.dll		Uncomment only this line	Commented
RM_ACS	Number	ACS (Automated Cartridge System) controlled by the Robot Manager module.	0

The following table identifies common Sony PetaSite options:

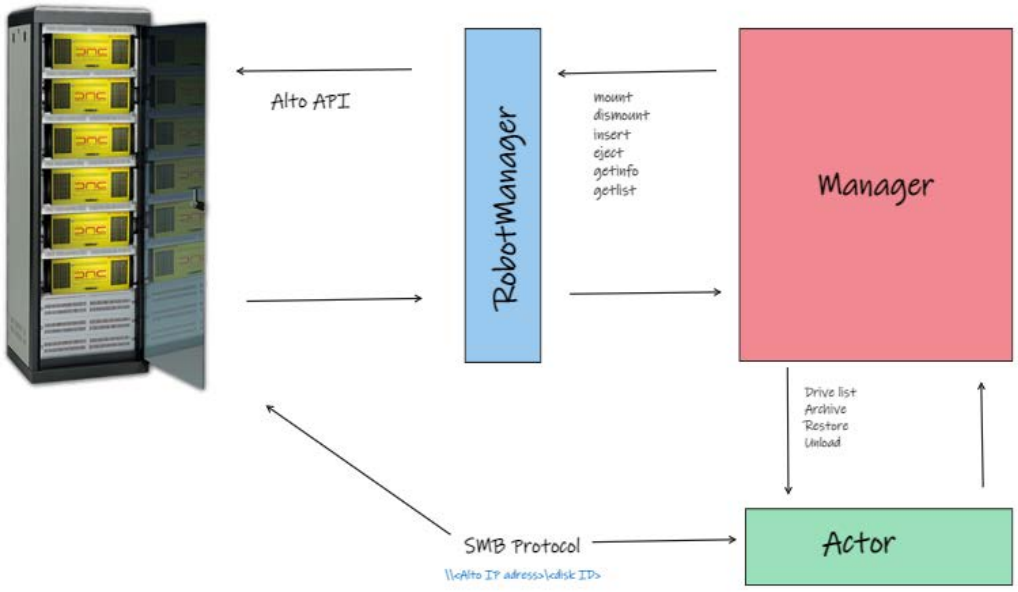
Parameter	Parameter Type	Description	Default
RM_SONY_ENABLE_MEDIA_TYPE_TRIMMING	Number	This parameter must not be modified during production. The database may need to be patched if it is changed during production. Some tape labels contain and additional two or three characters identifying the type of media. For example, 004452L2 is an LTO2 tape and S1000052 is a SAIT1 tape. If this parameter is set to 1, the Sony Robot detects the tape using the label and filters out the two or three additional characters from the label.	1
RM_SONY_MEDIA_TYPE_TRIMMING_LEFT	Number	This parameter must not be modified during production. The database may need to be patched if it is changed during production. Depending on the label, the two characters may be on the right or on the left of the label. Set this parameter to 1 if the Media Type information is on the left, otherwise set it to 0.	0
RM_SONY_PSCSERVERNAME	IP Address or Host Name	This parameter specifies the Host Name or IP Address of the Sony PSC (PetaSite controller). If you specify a Host Name, this must be defined in the operating system's hosts file.	

Parameter	Parameter Type	Description	Default
RM_SONY_PSCUSERID	Number	This specifies the User ID that the Robot Manager uses when it connects to the Sony PetaSite Controller.	1
RM_SONY_PSCTIMEOUT	Time in milliseconds	Command time out to the PSC in milliseconds. This is only used for mount operations.	900000
RM_SONY_PSCDISMOUNTRETRIES	Number of retires	The maximum number of retries when the dismounted drive is still in use. If the setting is 5, the initial delay is five seconds. The delay is then doubled after each retry.	5

Alto Disk Archive Library Configuration

Alto Disk Archive is a type of library of disks. Instead of mounting or dismounting tape, this library is designed to mount or dismount the file system of un-pluggable disks and map them to a SMB share.

DIVA supports Alto like a tape library. A disk of the library is seen as a tape in DIVA with its UUID as label.



For mount operations, RobotManager needs to use the Alto API to mount the disk and make it available via SMB with these default credentials:

login: *alto*

password: *password*

The Eject command supports disk export functionality. It marks the disk as removed, so the subsequent operations on that disk aren't allowed until the disk is made available again.

The Insert command is not available with Alto.

Configuring Robot Manager

Set the following parameters in the Robot Manager configurations file:

- Set RM_MODULE to ALTO_Robot.dll so that RobotManager loads ALTO_Robot.dll.

These parameters use the typical settings:

  - Set SERVICE_NAME to the name of RobotManager.
  - If necessary, modify RM_PORT so that RobotManager listens on different TCP port.
  - RM_ACS can be used to setup multiple Alto servers in the same configuration.
- RM_ALTO_SERVER_HOSTNAME must be set to the IP address of the Alto server.
- RM_ALTO_SERVER_PORT is the TCP port of the Alto API. The default value is 6480.
- RM_ALTO_GROUP can be set to only use a specific group of disks with DIVA. If this parameter is not set, or empty, RobotManager will use the group named "default".
- RM_ALTO_GROUP_CREATE_IF_NOT_EXISTS can be set to 1 and then RobotManager will create the group at startup. This parameter is set to 0 by default.
- RM_ALTO_NUMBER_OF_DRIVES represents the number of virtual tape drives. This is the virtual entity for the Alto integration, but is required by DIVA because tapes are mounted into drives. This number also defines the number of disks that DIVA can mount at the same time.

Actor AltoVirtualDrives Configuration

AltoVirtualDrives contains a pattern string so that an actor can build the list of ALTO Virtual Tape Drives during scandrive for each ACS. To configure ALTO Virtual Tape Drives, browse to
Configuration > *SYSTEM SETINGS* > *ACTORS* > *EDIT ACTOR* > *Advanced Settings* tab.

Syntax:

```
<pattern1>[,<pattern2>][…]
```

Pattern syntax:

```
<alto-hostname>:<alto-port>-ACS<acs number>-<number of virtual
drives>
```

Example:

```
192.168.59.102:6480-ACS1-2,192.168.59.102:6480-ACS0-10
```

Simulated Media Changers Configuration (for DIVA Simulators)

Simulated robots are available on Windows platforms. The settings are shown here for reference only. Refer to the DIVA Simulator Operations Guide (available to OPN partners only) for more information on installing and configuring a DIVA Simulator platform.

robotmanager.conf Common Options

The following table identifies common robotmanager.conf options:

Parameter	Parameter Type	Description	Default
RM_MODULE=SIMULATOR_Robot.dll		Uncomment only this line	Commented
RM_ACS	Number	ACS (Automated Cartridge System) controlled by the Robot Manager module.	Ignored

The following table identifies the DIVA Simulator parameters:

Parameter	Parameter Type	Description	Default
RM_SIMU_BASEDIR	Directory Path	The DIVA simulation files base directory path. This is typically C:\Diva\Simulation.	
RM_SIMU_OPERATION_SHORT_DELAY	Time in milliseconds	This setting simulates physical delays in mount, dismount, enter, and eject operations. The recommended setting is 10000 msec.	0
RM_SIMU_OPERATION_LONG_DELAY	Time in milliseconds	You can use this setting to simulate an operation that takes more time than expected for execution. The recommended setting is 120000 msec.	0
RM_SIMU_OPERATION_LONG_DELAY_FREQUENCY	Number	This setting specifies how often a long delay should occur. The recommended setting is 50.	0

Parameter	Parameter Type	Description	Default
RM_SIMU_LIST_SHORT_DELAY	Time in milliseconds	This setting introduces a simulated physical delay in list operations. The recommended setting is 500.	0
RM_SIMU_LIST_LONG_DELAY	Time in milliseconds	You can use this setting to simulate a list operation that takes more time than expected for execution. The recommended setting is 60000 msec.	0
RM_SIMU_LIST_LONG_DELAY_FREQUENCY	Number	This setting specifies how often a long delay should occur in list operations. The recommended setting is 100.	0

Robot Manager Library Interface

After configuring the Robot Manager configuration file, launch the Core Robot Manager and confirm that the library itself can be controlled.

Library interfaces that use ACSLS or PSC intermediate control software must be running before launching the Core Robot Manager. ACSLS controlled Managed Storage should also be varied online (for example, vary lsm0 online).

Robot Manager Start, Stop, and Restart

Windows Core Robot Managers start automatically with Windows. You manage (start, stop, restart) the service through the Windows Services applet.

Note: If the library is offline when the service is started, the Robot Manager doesn't automatically reconnect after the library comes online. You must restart the service to connect.

A Robot Manager can also be stopped and then started (restart) from a command window. The quotation marks in the commands must be used when specifying a service with spaces in the name. Use the following command sequence to stop and then start the service:

```
net stop "Core RobotManager"
net start "Core RobotManager"
```

You use the following command sequence if the *SERVICE_NAME* is specified in the robotmanager.conf file:

```
net stop "Core RobotManager SERVICE_NAME"
net start "Core RobotManager SERVICE_NAME"
```

Robot Manager Library Control

Caution: Don't use these utilities in a live DIVA system. Don't send commands to a Robot Manager, using either of these utilities, when the Manager is running. Technical Support isn't responsible for any complications arising from inappropriate use of these utilities.

Either the Robot Manager Client (a command-line interface) or GUI can be used to establish basic control functionality of a Robot Manager to its controlled Managed Storage. You can use either of these utilities to send manual commands to a Core Robot Manager to initiate simple operations, for example, drive mounting, dismounting, enter or eject operations from the Cartridge Access Port (CAP). Both utilities connect to a Robot Manager through TCP/IP and can be run from a remote computer. This feature enables you to use the Robot Manager GUI from a remote computer.

If you mount a tape with either of these utilities, you must first unload the tape before it can be dismounted, unless the library supports Forced Dismount commands and they are enabled in the Core Robot Manager configuration file.

Robot Manager Client

This command-line client is typically located with the Robot Manager executable files in the `%DIVA_HOME%\Program\RobotManager\bin` folder.

You must specify the IP address of the Robot Manager and its TCP port when launching the client as follows:

```
RobotManagerClient {IP_Address} {TCP_Port}
```

The IP_Address is the IP address of the Robot Manager computer, and the TCP_Port is the Robot Manager listening port. You can hard-code these two parameters in the Robot Manager Client batch file if there is only a single Robot Manager requiring access.

All of the client commands are self-explanatory after you start the program.

Robot Manager Client GUI

The Robot Manager Client GUI is typically located with the Robot Manager executable files located in the `%DIVA_HOME%\Program\RobotManager\bin` folder. The GUI provides the same functionality as the command line client. To open the GUI interface run `RobotManagerGU.bat`.

The GUI interface includes the following buttons and functionality:

Connect Button

Click this button to connect to the Core Robot Manager. You must enter the IP address and TCP port of the Core Robot Manager to be tested in the *Connect* prompt.

Tape List Button

Click this button to load the tape list from the library.

Reload Config. Button

Click this button to reload the configuration.

Exit Button

Click this button to exit the program.

Tape List

To manually mount a tape, select a Barcode ID and drag and drop it on to one of the drives displayed in the LSM list.

LSM List

This area lists all of the available drives and the tapes in the drive. You right-click a tape and select *Dismount* from the menu to dismount a tape.

CAP List

To manually eject a tape from the library, select a Barcode ID and drag and drop it to one of the listed CAPs.

Status Area

The *Status* area is at the bottom of the view and displays status messages from the Robot Manager.

Robot Manager System-Level Configuration

At the system level, each instance of the Core Robot Manager must be declared to the Manager in the Robot Managers area of the Robots view in the DIVA Web App.

You use area buttons to add (+), edit (Edit), or delete (-) a Robot Manager. The Update area button refreshes the displayed Robot Manager information from the database.

Clicking the + button adds a Robot Manager to the configuration. The Add new row in Robot Managers dialog box is displayed. Enter the following information in the appropriate fields and then click OK to add a Robot Manager:

Name

The name of the Core Robot Manager attached to this DIVA system.

Address

The IP address of the host running the Core Robot Manager installation.

Port

The Robot Manager TCP port. This must match the RM_PORT parameter specified in robotmanager.conf.

Site

The Manager uses this parameter to determine optimal use of resources in resource allocation. Use the menu list to select the appropriate site for this Robot Manager. Site Selection must be enabled in the Manager configuration file or all sites are considered equally.

Robot Manager Activity Logs

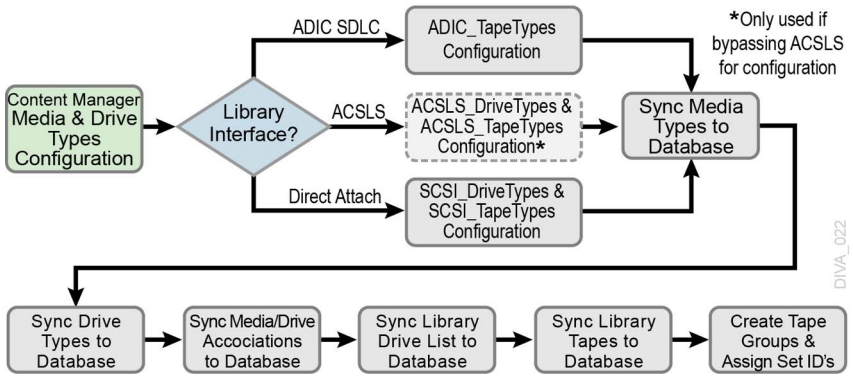
During normal operation, each Core Robot Manager logs its communications with the library and stores them in the `%DIVA_HOME%\log\robot_manager` folder. These logs are useful for troubleshooting issues. You may be asked to provide the log files when contacting Technical Support.

The most recent log file is named `robot_manager.log` or `robot_manager_SERVICE_NAME.log` and is located in the `..\log\robot_manager` folder. Older logs are renamed with the time it was saved as its file name and moved to dated sub-folders under the name of each Robot Manager.

Media and Drive Type Configuration

After you have successfully configured the Core Robot Manager for your Managed Storage, and the appropriate details for all Core Robot Managers entered into the Robots view section, the tape media, drive models, and the drive locations currently installed in each library must be entered.

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



Tape drives and their associated media types that are installed in a particular library are initially configured in the DIVA Database using static configuration files. The files are

located in the `%DIVA_HOME%\Program\conf\robot_manager` folder. The Core Robot Manager selects the appropriate files according to the RM_MODULE setting configured in `robotmanager.conf`.

The following list identifies the configuration file names and use:

scsi_drive_types.ini and scsi_tape_types.ini

Used for direct attached SCSI Managed Storage. These files are only considered if the INI extension is removed.

acsls_drive_types.ini and acsls_tape_types.ini

Used for Managed Storage managed by ACSLS. Normally, tape and drive types are derived from ACSLS during library synchronization with the database. However, you can use these files to override the values returned from ACSLS. These files are only considered if the INI extension is removed.

SCSI_drive_types and ACSLS_drive_types

You can edit these files using any plain text editor (for example, Notepad or Notepad++). No modification of these files is required other than to remove comment fields for the appropriate library and drive types for your installation.

Remove the # at the beginning of the line in the appropriate library section for the drives to be recognized in a Synchronize Drive Types List in DIVA. You must leave drive types in Managed Storage not installed commented out.

The Compatible Drive Types column cross-references the Tape Type ID in SCSI_Tape_Types (or ACSLS_Tape_Types if used). These values are examined in the Synchronize Media/Drive Compatibility List procedure in the DIVA Web App.

SCSI_tape_types and ACSLS_tape_types

You can edit these files using any plain text editor (for example, Notepad or Notepad++). No modification of these files is required other than to remove comment fields for the tape types for your specific library.

Remove the # at the beginning of the line in the appropriate library section for the tapes or DVDs to be recognized in a Synchronize Media Types List. You must leave tape types (or DVDs) in Managed Storage not installed commented out.

The Compatible Drive Types column cross-references the Drive Type ID in SCSI_Drive_Types (or ACSLS_Drive_Types if used). These values are examined in the Synchronize Media/Drive Compatibility List procedure in the DIVA Web App.

Tape Group Configuration

You must create tape groups before configuring the Storage Plans. The number of Storage Plans required is usually the same as the number of tape groups configured.

Tape groups enable DIVA to physically separate archived content into different tapes, and typically creating one Storage Plan per tape group is necessary. All Storage Plans are typically setup to be the same, except that the copy goes to different a tape group. However, the more tape groups DIVA uses, the less efficiently content will be stored across tapes.

Note: Refer to the DIVA Storage Policy Manager documentation for details on SPM configuration and use.

If complex objects are going to be used in the system, tape groups must be set up containing tapes configured for AXF format. Complex objects aren't compatible with the Legacy formatted tapes or disks. Non-complex objects can be stored using either Legacy or AXF format.

The following example illustrates how creating too many tape groups causes fragmentation of objects across the tape group. Using fewer tape groups results in fewer Storage Plans to setup, less fragmentation across tapes, and is easier to maintain.

Having fewer tape groups will resolve the issue in the following example, and avoid fragmentation across the tape groups.

Example Configuration:

- 10 Tape Drives
- 30 Tape Groups with SetID=10
- 300 Total Tapes assigned to SetID=10

Results:

- Each tape group (in the worst case) will use at least ten tapes, and store files on each tape when an object is archived.
- After objects are archived to all 30 tape groups, all 60 tapes (total) are used.
- Over time, if any of the tape groups is heavily archived, and one of the tapes is 100% full, no more tapes are available.
- DIVA will run out of tapes, even though a lot of the tapes are still mostly empty (each containing only one object), but can't be used because it was assigned to a different tape group.

Reasons for Creating Multiple Tape Groups

Create a multiple tape group only for a good reason. Multiple tape groups cause tape-group fragmentation, resulting in some tapes not being filled, and storage space being wasted.

Several valid reasons for creating multiple tape groups follow.

- Long and short form materials should be stored on different tape groups. If small objects are mixed with larger objects on the same tape, access to the smaller object will be delayed for an extended period until the larger object restore is complete.

- Content that is deleted regularly from the archive should be stored in a different tape group than content that will never (or rarely) be deleted. Deleting from tape will cause tape fragmentation and the fragmented space can't be used until the tape is repacked. If the two types of content are mixed together, deleting will cause more tapes to become fragmented, and repacking will be required more frequently and take longer.
- Online and backup copies of the same content should be on two different tape groups. Backup copies are meant to be removed from the tape library and stored in an Iron Mountain type of facility as a backup. However, if both copies are mixed in the same tape group, it is impossible to determine which tape contains the backup copy. The result is being unable to remove it from the library for offline storage.
- When requirements necessitate using tapes purchased by different departments and enforcing that each department uses only the tapes they purchase themselves, tape groups must be created for each department with a different set of tapes.
- Different storage formats must be assigned to different groups. For example, one group would be Legacy Format while another group is AXF Format. Complex objects aren't compatible with legacy format, and must be processed to and from AXF formatted medium. Non-complex objects are compatible with both Legacy and AXF formats.

Invalid Reasons for Creating Multiple Tape Groups

Too many tape groups cause DIVA to work less efficiently, result in tape group fragmentation, and are more difficult to configure and maintain. Use as few tape groups as possible.

See *Tape Group Configuration* for recommended best practices.

The following are several invalid reasons for creating multiple tape groups:

- To store different content in different tape groups because you think it is easier to manage. DIVA manages the tapes. When restoring an object on a tape, DIVA automatically knows which tape the object is on, and mounts that tape, or notifies the user to insert the necessary tape into the library. It will not require the user to figure out which tape is needed.
- Creating many tape groups because it is easier to search. DIVA performs searches using object metadata stored in the database, not tape groups. The tape group only makes sure content is physically separated, and doesn't assist in searching functions.
- Creating multiple tape groups for cataloging. When migrating from an analog tape environment, you might label what is recorded on each analog tape directly on the tape itself, and then, store a group of the tapes on different sections of a shelf.

Caution: DIVA doesn't work efficiently this way; this method should be avoided.

Tape Drive Reservations

While DIVA uses a drive for an archive, a restore, or a copy, this drive must not be used by another request or another software (such as Tape Reading Utility). You must restrict access to tape drives to prevent concurrent operations that would be sent to a tape drive from different initiators.

When an actor reserves a drive, the actor must supply the drive with a reservation key composed of the actor name (the CRC 32 of the actor name) and the request ID.

If two restores use the same tape drive at the same time, these restores often abort, or restore corrupted data. The worst-case scenario is concurrent access to a tape drive in the middle of an archive request. For example, when a tape is rewound by another initiator while DIVA is archiving to that tape, the content of the tape is lost. DIVA Manager prevents a drive from being used by more than one request at a time, but this doesn't prevent actors from reserving drives too.

For each transfer involving a tape drive, actors perform drive reservations using a SCSI command: persistent reserve out. A drive is reserved at the beginning of a transfer and released at the end. Each reservation is associated with a 64-bit key: the reservation key. When an actor reserves a drive, the actor must now supply DIVA with the actor name and request ID in a CRC32 string.

For example, the CRC32 of actor1 is 47f9d059, so the first 32 bits of all the reservation key issued by actor1 is set to 47f9d059. The last 32 bits of a reservation key are set to the ID of the request that reserved the drive. The last 32 bits of a reservation key are set to the ID of the request that reserved the drive.

There are online tools to compute CRC32 from a string like CRC. Before making a reservation, actors verify that the drives are not already reserved. If a reservation conflict is detected by an actor, an error message containing the current reservation key is sent to Manager and added to the request log.

Example: Drive is not accessible [Reservation Conflict, the drive 37D12EF048 is already reserved by ReservationKey=47f9d05900000133(crc=47f9d059, requestid=307)], drive[Tape0,IBM-LTO6]

The following help identify what locked the drive:

- Serial number of the drive that is already reserved
- The ReservationKey:
  - CRC of the actor name that made the reservation
  - Manager request ID

If an actor crashes in the middle of a transfer, it doesn't release the reservations already made; and subsequent requests might get a reservation conflict. In the example, the actor that locked the drive is actor1 (the crc32 of actor1 is 47f9d059) during the execution of the request 307. From a given CRC32 value, the only way to come out with the actor is to compute the CRC32 of all the actors of the system.

- The SC (Scandrive)

Scandrive is now embedded in `ActorService.exe`. This version of scandrive can also report the reservations, as in this example:

```
C:\DIVA\Program\Actor\bin\ActorService.exe scandrive
Scsi1:0:1:0
        Type : TapePeripheral
        Identifier : HP      Ultrium 6-SCSI  HB91
        DeviceName : Tape1
        Serial number : A496500001
        Firmware version : HB91
      Reservation : ReservationKey=47f9d05900000133(crc=47f9d059,
requestid=307)
Scsi1:0:4:0
        Type : TapePeripheral
        Identifier : HP      Ultrium 5-SCSI  6.24
        DeviceName : Tape0
        Serial number : A777C00603
        Firmware version : 6.24
        Status : No media in drive.
```

In case a drive must be unlocked manually, scandrive can also be used in order to clear a reservation. This operation must be executed on the host that made the reservation. Before clearing a reservation, you need to ensure that the request that owns the reservation is no longer running. Run this command to clear a reservation from a tape drive: `ActorService.exe scandrive -release <drive serial number>`.

Note: Actors automatically clear their reservations on startup in case a reservation was left; for example, after a crash.

When the `-release` option is set, scandrive still reports the list of drives. If the drive with the specified serial number is reserved, scandrive clears it. Scandrive can be executed a second time to make sure the reservation is actually cleared, as in the following example:

```
C:\DIVA\Program\Actor\bin>ActorService.exe scandrive -release
A496500001
Scsi0:0:0:0
        Type : DiskPeripheral
        Identifier : NVMe    KBG30ZMT512G TOSADLA
        DeviceName : PhysicalDrive2
Scsi1:0:1:0
        Type : TapePeripheral
```

```
        Identifier : HP      Ultrium 6-SCSI  HB91
        DeviceName : Tape1
        Serial number : A496500001
        Firmware version : HB91
      Reservation : ReservationKey=47f9d05900000000(crc=47f9d059,
requestid=0)
        Reservation cleared
Scsi1:0:4:0
        Type : TapePeripheral
        Identifier : HP      Ultrium 5-SCSI  6.24
        DeviceName : Tape0
        Serial number : A777C00603
        Firmware version : 6.24
        Status : No media in drive.
```

Tape Label Management

When a tape is first mounted and objects are written to it, DIVA writes a label to the beginning of that tape. The label contains important information relating to the management of objects written to, or deleted from, the tape during archive operations. From an operational perspective the most important information in the tape label is the barcode number of that tape. The barcode is an alphanumeric number on the physical label adhered to the back of the cartridge, and is also written to the label on the magnetic media of the tape.

DIVA automatically checks the label written on tape whenever a tape is mounted to verify that it matches the tape barcode label it instructed the tape library to mount.

This mechanism provides the following two safety features:

- Confirmation that the mapping between the physical drives in the library matches that of the logical connections to each tape drive from the actor. This prevents data being written to the wrong tape if there is a configuration mismatch between the physical drives in the library.
- Prevents tapes with foreign labels (that is, tapes previously used by another archive system) from being overwritten in error. This behavior is for environments where DIVA shares a library with another archive application and the tapes used by that archive application have not been set to Set ID 99.

If DIVA identifies a mismatch between the expected label and that of the tape, it generates an I/O Label Error, and the tape is set to Not Writable and isn't selected for any further write operations.

Tape Drive Encryption

Enable or disable tape drive encryption in the DIVA Web App on the *EDIT TAPE GROUPS* view. Tape drive encryption supports secure bulk tape migration between DIVA systems.

If encryption hasn't been enabled—or if encryption has been disabled—on a tape group, any tapes in the tape group are unencrypted. Once encryption has been enabled on a tape group, all tapes added to the tape group are encrypted, but tapes that were in the tape group before encryption remain unencrypted.

Enabling encryption on a tape group generates an encryption key, which also is encrypted. You can change the encryption key at any time through the DIVA Web App. Updating the encryption generates a new key.

New tapes added to the tape group after encryption use the new encryption key. The tapes in the tape group that already were encrypted continue to use the original key. Therefore, tapes in the same tape group can have different encryption keys. Manager must be notified of the change when updating the encryption key.

The encryption status of the tape can be viewed on the *Resources Management* > *Tapes* view by clicking on the tape barcode to display the *TAPE PROPERTIES* view.

Enabling Tape Drive Encryption

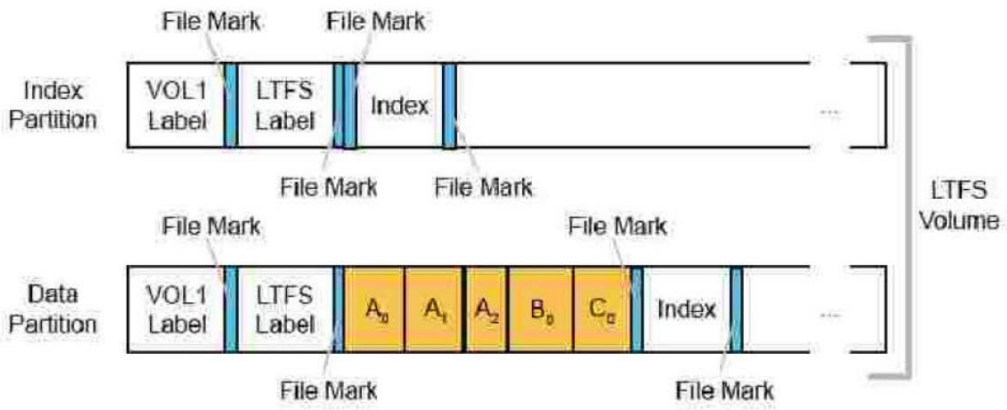
To enable, disable, or update tape drive encryption, do the following:

1. Click the *Filter* button.
2. Browse to *Resources Management* > *Tapes*.
3. Open the *TAPE GROUPS* view.

   For detailed configuration, export, and import information, see the *DIVA Installation and Administration Guide* at https://www.telestream.net/telestream-support/diva/support.htm.

AXF LTFS and TAPE Encryption

DIVA supports encryption with AXF LTFS. When encryption is enabled, DIVA encrypts the data in the orange blocks shown in the following diagram. The other blocks containing VOL1, LTFS label and LTFS index are not encrypted.



To read an encrypted AXF LTFS tape using third-party LTFS software, the encryption must be enabled in mixed mode.

Contact Telestream support for guidance on third-party tools that can encrypt tapes.

Obtaining the Encryption Key for an Encrypted Tape

To obtain the encryption key for a tape, do the following:

1. Log into DIVA in as either a System Administrator or Administrator.
2. In the DIVA Web App, browse to *Resource Management* > *Tapes*.

   DIVA displays a list of tapes.

Note: The encryption key is also available from the tape property page in the action menu.

---

3. Open the action menu for the tape for which you want the encryption key.
4. Select *Get Encryption Key*.

   DIVA opens a modal form displaying the encryption key for the tape.
5. To copy the encryption key to the clipboard, click the Copy to Clipboard button to the right of the encryption key.

   If the tape isn't encrypted, DIVA displays the message, *No Encryption Key Available*.

Tape Compression

Tape compression is supported at the tape group level. When tape compression is enabled, any empty tape assigned to the tape group has compression enabled, and

instances written to the tape is compressed. Tapes assigned to the tape group before compression was enabled remain uncompressed, and instances written to the uncompressed tape is uncompressed.

When writing to a tape group with a compression setting (enabled or disabled), only used tapes with the same compression setting, or empty tapes associated with the set linked to the tape group, are used.

When exporting a tape, compression is tracked using the `isCompressionEnabled` attribute. This attribute value can be either true or false.

Viewing Compression-Enabled Tapes

Tape compression is supported at the tape-group level. When tape compression is enabled, any empty tape assigned to the tape group has compression enabled: instances written to the tape are compressed. Tapes assigned to the tape group before compression was enabled remain uncompressed, and instances written to the uncompressed tape are uncompressed.

To view all tapes with compression enabled, do the following:

1. Browse to *Resources Management* > *Tapes*.
2. Open *Filters*.
3. Set the *Compression Enabled* filter to *Yes*.
4. Click *Apply Filters*.

Editing Tape Groups

To edit tape groups, browse to
Configuration > *Library Storage* > *Tape Groups* > *Edit Tape Groups*.

Tape Spanning

When the contents of a tape group begins to reach full capacity, and the Set ID associated with the tape group has no more blank tapes to draw upon, DIVA maximizes the storage capacity by spanning files across tapes.

By default, the tape spanning feature is configured in the DIVA Configuration file to *not* span tapes. If an object can't be spanned across the remaining free space of two tapes within that tape group, the job is terminated by DIVA. Tape spanning can be configured to span across more than two tapes at the site, or disabled altogether in the diva.conf configuration file.

During the restore of a spanned object, DIVA mounts all associated spanned tapes and automatically joins the spanned object back together. It can't do this directly, and must copy all segments of the spanned file to a cache disk first. Therefore, restoring a spanned object must use a *Cache Only* or *Cache and Direct* QOS. A Direct QOS results in the job terminating.

For Write-Once media, objects are spanned when there is 100 MB remaining so that there is space left and the disc can be finalized. After an object is spanned the disc is considered full and is automatically finalized.

If spanning is disabled and an object is to large to fit on the selected tape, DIVA retries to fit all content on a single empty tape. The retry logic can be configured to optionally retry with a used tape with the least or most remaining space. The type of tape DIVA selects on retry is configurable in the DIVA configuration file through the `DIVA_RETRY_ON_SPAN_REJECTED_ALGORITHM` configuration parameter.

Note: Any job that writes to a tape linked to a clone, terminates on a span. In addition, any attempt to clone a tape with spanned instances aborts.

True Remaining Tape Size and Last Written Position

For some specific tape drives (Oracle T10K and LTO) the actor returns the True Remaining Size on the tape and the Last Written Position on tape to DIVA during a transfer of content to tape. The remaining size is given in number of bytes of uncompressed data.

DIVA uses the remaining size and last written position (instead of relying on the size of the tape type) to obtain the true total and remaining size on the tape in all tape based operations.

Export and import operations also now include the total tape size.

Repacking Tape Jobs

The Repack Tape job sends a repack job for the selected or specified tape. A tape repack operation reclaims unusable space on a tape due to object deletions, and removes fragmentation.

Caution: The Repack Tape function is not intended to move material from a tape that is already known to be generating read errors. Contact Telestream.

Use the following process to repack a tape:

1. Browse to *Resources Management* > *Tapes*.
2. Click the three dots next to the barcode of the tape to be repacked, or select the check box next to the tape(s) to be repacked.
3. Select *Repack Tape* from the menu, or click the *Repack Tape* button at the bottom of the tapes list (if the check box method was selected).
4. Confirm the barcode on the form and select the job priority.
5. To execute the job, click *Submit*.

Tape repacking can be a lengthy process and DIVA, by default, considers tape repacks a low priority operation. If higher priority jobs are issued, a tape repack job can be temporarily suspended while the higher priority jobs are completed. If higher priority jobs are issued sporadically it can result in frequent mount and dismount operations of the drive performing the repack. Therefore, Telestream recommends that tape repack operations should be run during off-peak periods where the frequency of higher priority jobs is limited. Some installations may have a drive dedicated solely to tape repacking to prevent this scenario from occurring.

The repack process involves the following tasks (in order):

1. Mounting the source tape and reading all objects to temporary disk cache of an actor enabled for repack operations.

Note: If the temporary disk cache is filled before reading all objects from the source tape, DIVA proceeds to Step 2 until the cache is cleared. DIVA then proceeds to read the remaining objects from the source tape. This process is repeated until all objects are read.

---

2. Mounting a tape from the Unused Tapes Sets pool associated with the Set ID of the tape group from the source tape.
3. Writing all objects from the temporary cache in Step 1.
4. Deleting the objects from the temporary cache after all objects have been successfully written to the new tape.
5. The original source tape is released to the Unused Tapes Sets pool and unassigned from the tape group.

If a read error occurs at some point during the repack process from the source tape or a write error occurs on the destination tape, the entire repack job is terminated and no objects from the source tape are deleted. If the cache filled during the repack job and objects were successfully written to another tape before the cache was cleared, those objects remain on the destination tape.

If a read error occurred, the source tape's repack status and write status is disabled. If a write error occurred, the destination tape has its write status disabled and won't be used for any tape write operations. The write and repack states of both tapes is viewable on the *Content Management* > *Job History* view.

During the manual repack of WORM media, the usual dialogs display, but a warning is included notifying you that the space on the source media isn't recoverable after the repack is complete. Deletes and Repacks do not clear WORM media because these are Write-Once media. The instances are deleted, but the space is not recoverable.

Use the same process previously described to also repack WORM media. The following fields are included in the Repack Tape form:

- **Repack WORM (Write-Once media) with barcode**

  Barcode of the tape to be repacked.

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the Priority Value text box.

  There are six predefined values as follows:

  - `Min`
  - `Low`
  - `Normal`
  - `High`
  - `Max`
  - `Default`

  If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

Cloned Tape Repack

Tapes linked to a clone can't be repacked. If a repack is attempted on a Source or Clone Tape, the job terminates. The clone Storage Link must be removed to repack either tape.

Repacking Tapes Automatically

The Automatic Repack option allows configuring scheduled tape repack operations. Use the following process to enable automatic tape repacking:

1. Browse to the *Resources Management* > *Tapes*.
2. Select a tape, or tapes, using the check box next to the tape barcode.
3. Click the *Automatic Repack* button at the bottom of the tape list.
4. Select the *Automatic Repack Enabled* check box and fill in the information on the form.
5. To enable automatic repacking of the selected tape(s), click *Submit*.

Verifying Tape Jobs

The *Verify Tape* job initiates a system read-back through every object on the selected tape one at a time and verifies all of the checksum values.

To verify a tape or tapes do the following:

1. Browse to *Resources Management* > *Tapes*.
2. Check the box next to the tape(s) to be verified.
3. Fill in the required information in the form.
4. To execute the job, click *Submit*.

The following fields are included in the Verify Tape form:

- **Verify Tape with barcode**

  Barcode of the tape to be verified.

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the Priority Value text box.

  There are six predefined values as follows:

  - `Min`
  - `Low`
  - `Normal`
  - `High`
  - `Max`
  - `Default`

  If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

-

Tape Eject/Insert

Ejecting Tape Jobs

The Eject Tape job ejects the selected tapes from the associated library. One or more tapes can be selected simultaneously using the check boxes next to the tape barcode. To eject a tape or tapes, do the following:

1. Browse to *Resources Management* > *Tapes*.
2. Check the box next to each desired tape.
3. Click the three dots and select Eject Tape(s) from the menu, or click the Eject Tape(s) button at the bottom of the tapes list.
4. Fill in all fields in the form displayed.
5. To execute the job, click *Submit*.

The following fields are included in the Eject Tape(s) form:

- **Comments**

  Comments can be added when the tape is ejected. These may refer to its location or other information. Comments can be viewed later by examining that tape's properties in the *Resources Management* >*Tapes* view (click the tape barcode to view the tape properties).

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the Priority Value text box.

  There are six predefined values as follows:

  - Min
  - Low
  - Normal
  - High
  - Max
  - Default

  If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

Cloned Tape Eject

Ejecting cloned tapes works in the same way as exporting them; the associated tape is ejected as well. To disable this behavior, remove the clone storage link. Then, if only the clone tape is ejected, the source Tape must be set to not writable.

Protected Mode

When a tape is ejected from the library it is automatically set to Protected Mode. When this attribute is set it prevents further archive operations from being performed on the tape and prevents the tape from being repacked.

DIVA assumes that when a previously-ejected tape is reinserted into a library to perform a restore operation it subsequently is ejected and put back into offline storage. Without the Protected Mode feature new virtual objects may be written to the tape while it is temporarily in the library and prevent it from being ejected without first moving these required objects to another tape.

Write operations on a protected tape aren't allowed unless the protected attribute is set back to false in the DIVA Web App after the tape is reinserted into the library. This attribute doesn't prevent delete operations on instances located on these tapes (whether internalized or externalized).

This attribute may also need to be reset on a tape if the tape was mistakenly ejected from a library, or if the tape was stuck in a tape drive and removed by opening the library door and manually ejected from the library. When the library is then resynchronized with the Core database, the missing tape is deemed externalized and Protected Mode set to true (the tape is protected).

Inserting a Tape Into a Library

The Insert Tape job enables inserting a tape into a library through its CAP. The tapes can only be entered in the CAP after this command is issued with some library configurations.

Initiate an Insert Tape job using the Insert button located under the tapes list on the *Resources Management* > *Tapes* view.

The Sony PetaSite PSC software enables you to enter a tape in its CAP and manually place it within the PetaSite. In this case, DIVA is not informed of the action by the PSC and doesn't recognize the tape until the library is audited using the DIVA Web App.

To insert a tape into a library, do the following:

1. Browse to *Resources Management* > *Tapes*.
2. Click *Insert* at the bottom of the tapes list.
3. Fill in all fields in the form.
4. To execute the job, click *Submit*.

The following fields are included in the Insert Tape form:

- **Robot Manager Name**

  This list specifies the Robot Manager controlling the associated library for insertion of the tapes. Use the menu to select the appropriate robot manager.

- **CAP ID**

  This list is for Managed Storage with multiple CAPs. Some Managed Storage doesn't unlock the CAP, enabling the tape to be inserted, until the Insert Tape command is issued. You can specify which CAP to unlock from this menu.

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the Priority Value text box.

  There are six predefined values as follows:

  - `Min`
  - `Low`
  - `Normal`
  - `High`
  - `Max`
  - `Default`

  If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

Export and Import Functions

The Export function (on the first DIVA site) generates metadata files that describe each tape selected for export, and then ejects the selected tapes from their current tape library.

All export functions and the Insert Tape command are executed from the DIVA Web App. The Import Tape function uses the command-line interface. DIVA enables more than one set of tapes (whether spanned or not) to be exported to and imported from a single file.

The Import function is used to import the metadata, and then insert the ejected tapes into the second system. The archived objects on the exported tapes are then transferred to the second DIVA system.

Newly imported objects have only one instance—the instance residing on the tape(s) that was imported. The option is available to import an object as an instance of another object already existing in the Core database. The Import Utility requires specification of a target tape group for newly imported tape objects. The new objects belong to the identified tape group and not the tape group of the DIVA system from which it was exported.

The Export/Import functionality is compatible with complex objects and has additional fields for the advanced formatting and functionality available in DIVA.

Note: The exported metadata from a DIVA export can't be imported into DIVA releases before 7.0. However, exported metadata created from releases of DIVA before 8.3 can be imported into DIVA.

Tape Export

The Export Tapes function enables one or more tapes containing objects to be exported for use in another independent DIVA system (for example at a remote disaster recovery or partner site).

The metadata of each tape for non-complex objects are maintained in the Core database. The metadata of each tape is saved to an XML file when the tape(s) are exported and used to transfer the metadata to the other DIVA system's database during the import operation.

The metadata for complex objects is maintained in both the Core database and the Metadata database. When an export job is initiated, the Export Utility creates an additional plain text file and assigns an FFM extension to the file.

The export feature checks to see if any of the selected tapes contain objects that span onto other tapes. If so, these tapes are included in a menu so that they can also be exported. These spanned tapes must be selected to export the original list of tapes.

The Export Tapes command is not used for transferring tapes between two or more Managed Storage controlled by the same DIVA. To transfer tapes between Managed

Storage under the same DIVA control, use the Eject command, move the tape to the desired library, and then execute an Insert Tape command.

The default action in the export feature removes the tape metadata from the Core database after the export. In this case, if an object being exported is the last (or only) instance of the object, it is removed entirely from the database. However, the object metadata can be left in the original Core database if desired.

Ejected tapes can also be exported. Ejecting tapes before exporting them is the recommended method when the number of tapes to be exported exceeds the robotic tape library selected CAP (Cartridge Access Port) size.

The media type (Write-Once or not), and whether the media is a cartridge or not, is identified in the exported XML file and also imported during an Export/Import operation. The attributes of the tape element are `isWriteOnce` and `isCartridge` each with a value of either true or false.

Export Limitations

Tape export limits are configured in the diva.conf configuration file. There are several configurable parameters as described in the following table.

Parameter	Definition	Limits
DIVA_MAX_EXPORT_TAPES	The maximum number of tapes allowed in an export job. Re-loadable in SERVICE mode.	The range of possible values is 1 to 1000. Example: DIVA_MAX_EXPORT_TAPES=10
DIVA_MAX_EXPORT_ELEMENTS	The maximum number of elements allowed in an export job. Re-loadable in SERVICE mode.	The range of possible values is 1 to 10000000. Example: DIVA_MAX_EXPORT_ELEMENTS=1000000

Telestream highly recommends:

- Only performing one export operation at a time. You risk data loss if more than one export operation is running simultaneously.
- Not performing large exports during peak periods. System performance is decreased during large exports.
- Delete and repack actions do not clear WORM drives as these are Write-Once Media. The instances are deleted but the space is not recoverable.

Encrypted Tape Export

An encryption key hash and salt are generated during the export of encrypted tapes. This key hash and key salt are stored in the metadata file. The export process optionally generates a new password-protected Keystore file in the same folder as the metadata file. The encryption setting is turned on or off for a tape group on the *Edit Tape Group* view. You must be a System Administrator or Administrator to access the setting.

See *Editing Tape Groups* and *Tape Drive Encryption*.

The Keystore file contains information used to import encrypted tapes.

Note: A valid Keystore password is required to export encrypted tapes. For more information, see the following topic.

Export Keystore

The integrity of the Keystore file can be verified using the Java key tool. See https://docs.oracle.com/javase/6/docs/technotes/tools/solaris/keytool.html for details on Java's key tool.

The Keystore password is set on the *Configuration* > *General Settings* > *Security* view. Enter the Keystore password in the Export: Tape Encryption Keystore Password field. The password must be at least eight characters and contain at least one digit, at least one lowercase alphabetic character, at least one upper case alphabetic character, and at least one special character within a set of special chars (!@#%$^).

Enable exporting encryption keys by sliding the Export: Enable Export of Encryption Keys slider to On (slider background turns blue). Exporting encryption keys is disabled by default. You must be a System Administrator or Administrator to view or edit both settings.

Export Metadata Parameters

This table describes the export metadata parameters.

Parameter	XML Element and Attribute	Notes
objectId	Attribute of the object element	Not imported—a new object ID is generated during import.
uuid	Attribute of the object element	Imported if present, otherwise a new UUID is generated.
format	Attribute of the object element and attribute of the tape element	0 = Legacy 1 = AXF 0.9 2 = AXF 1.0 -1 = Unknown
numFolders	Attribute of the object element	

Parameter	XML Element and Attribute	Notes
isHeaderValid	Attribute of the object element	
isComplex	Attribute of the object element	
footerBeginPos	Attribute of the element's element	If exists in the database
footerEndPos	Attribute of the element's element	If exists in the database
compOrderNumBegin	Attribute of the element's element	If exists in the database
compOrderNumEnd	Attribute of the element's element	If exists in the database
fileFolderMetadataInfo	Element	Valid for complex objects
fileFolderMetadataInfo-elem	Element	Valid for complex objects
checksums and checksum	Element	Not valid for complex objects
elementIds	Attribute of the component element	The fully qualified path of Element ID values for a file or an empty folder's fully qualified path.
type	Attribute of the component element	Represents the type of object component: D = Directory F = File Components of non-complex objects created before the 7.4 release default to F because only files were stored in non-complex objects before release 7.4.

Exported Tape Metadata Files

When tapes are exported from the DIVA system, DIVA writes each tape's metadata to an XML file. DIVA generates an additional FFM file for each exported complex object. If an object is spanned across two (or more) tapes, the XML file encompasses every tape included in the spanned set. The naming format of each tape metadata XML file is Tapeset-<Barcode>.xml (for example Tapeset-000131.xml).

The Root Path where the XML files are saved is defined by the DIVA_EXPORT_ROOT_DIR parameter in the DIVA configuration file. By default the export absolute folder Root Path is `%DIVA_HOME%\Program\Core\bin\exported\`.

From this Root Path the XML and FFM files (if complex objects exist) from each Export Tapes command are saved in sub-directories based on the date and time the command was run.

The FFM file contains file and folder information for complex objects. The FFM files are referenced from within the specified XML file and are named using the object name and object collection of the exported object. This file must exist in the same directory as the XML file when importing. The Import Utility looks for them both in the same location. If the file is missing, the import process terminates and an error message is written to the log file.

Exporting Tapes

Caution: When using complex objects, the FFM files must be in the same folder as the XML files for importing. If the FFM files aren't found the import process terminates and an error is written to the log file.

The Export Tape job is initiated using the three dots next to the tape barcode, or by selecting the tape's check box and then clicking the *Export Tape* button at the bottom of the tape list. When selecting the tapes for export, it is possible to see more tapes available in the tape window than initially selected. If a tape has objects that are spanned onto another tape, these tapes are also included. In this case, select all of the spanned tapes from this list for the export to succeed.

To export tapes, do the following:

1. Navigate to *Resources Management* > *Tapes*.
2. Click the three dots next to the tape barcode, or select the tape's check box.
3. Select *Export Tape* from the context menu, or click the Export Tape button at the bottom of the tape list to begin the export process.

   The Export Tape dialog box displays, showing information about the selected tapes and options for the export process. Options include:

   – Comments

     Enter any comments desired in the text box. They are stored in the job's properties.

   – Delete From DB

     If checked, the barcodes, tapes, and object instances stored on those tapes are deleted from the Core database upon completion of the export. This parameter is set to false by default.

     If tapes or object instances are needed in the system again after they have been exported, they must be imported again because this option removes them from the database.

   – Exported Tapes

     This area identifies which tapes were selected for export, if the tape has the original barcode, and if it can be removed from the export operation. For example, if a tape is part of a tape set (rather than a single tape), the Can Be Removed column would indicate No for that tape because it is required to complete the export successfully.

Clicking the arrow on the left side of the tape in this area reveals whether there are objects on the selected tape.

– Remove Selected

Removes the highlighted tapes in the Exported Tapes area from the export process.

4. After all options have been set and verified, click Submit to begin the tape export.

This is a multi-step process. If a set of tapes was selected that includes another spanned tape, the GUI displays re-selection dialogs enabling selection of additional tapes in the set.

When you click the Submit button, the export process begins. This results in an XML (and possibly FFM files) being created in the export folder. The XML and FFM files contain all of the information concerning the objects on the tape(s) being exported.

When the export is complete, a good practice is to compress all of the resulting files into a ZIP file. Be sure to include all of the files because they are required for the import process to complete successfully.

Cloned Tape Export

Exporting a Source or Clone Tape triggers the export, also, of the associated tape. For example, if you archived content to source tape 3L2042, and cloned the contents to tape 3L204; when exporting source tape 3L2042, DIVA automatically includes Clone Tape 3L2048 in the list of tapes to export. Similarly, exporting Clone Tape 3L2048 triggers also the export of tape 3L2042.

The resulting exported XML file(*s*) contains a new element named `cloneBarcode`. It is used during import to restore the clone Storage Link. It also has the `lastCloneDate` for the Source Tape that is also restored during import.

Bulk Tape Export

DIVA internally splits an export job into smaller exports consisting of no more than 100 tapes and 1 million tape elements. This process enables DIVA to accept a larger number of tapes and tape elements than can be handled by earlier DIVA releases.

The number of tapes an export job is split into is always 100, and the number of tape elements are always 1 million. However, these values are adjustable.

For example, if you set the maximum number of tapes value to 3, the resulting export is split into smaller exports each including no more than three tapes. Each internal export generates a single XML file. All files are output to the same directory.

Tape Import

Importing tapes to be used in restore operations is a two-step process. First, the metadata that describes the tape objects is imported using the `importtapes` command line utility. After the metadata has been successfully loaded, the physical

tapes can be inserted into the tape library using the Insert function in the DIVA Web App.

Note: Multiple simultaneous import operations are enabled, but not recommended.

The Import Command

To use the `importtapes` command, first ensure that the exported XML metadata file and the FFM files exist on the destination DIVA System. The files must exist in uncompressed form in the DIVA bin directory (by default). Also, the object tape group must already exist on the target system before the import begins. This tape group doesn't necessarily have to be the same tape group assigned to the tape in the source system.

The associated XML files from exported tapes must be copied to the DIVA bin directory on the platform where they are to be imported. Before inserting the tapes into the library, the tape metadata must be imported into the Core database. This must be executed for each tape (or spanned set) to be imported.

Importing Tapes

Importing of tapes is accomplished using a combination of the Windows command-line interface and the DIVA Web App. Inserting the tape is an optional part of the workflow but is necessary to access the objects on the tape. It is possible to run the *importtape* command line utility to enter the tape's metadata into the Core database and still keep the tape externalized. However, to access the objects on the tape, the tape must be inserted using the Insert Tape function.

To import tapes into DIVA, do the following:

1. Open a Windows command-line interface.
2. Copy the exported XML and FFM Files into the `%DIVA_HOME%\Core\bin` folder.
3. Change to the `%DIVA_HOME%\Core\bin` folder.
4. Run the `importtape` command using any of the following necessary command line options:
   - `help` (`-h`)

     Displays help information.
   - `groupname`

     The tape group to which imported tapes belong. The tape group must already exist in the system.
   - `mfiledir`

     The XML file containing exported tape metadata, or a folder that contains the files.
   - `-skipIfNameExists`

     Skip import of objects with naming conflicts. The default behavior is that if the object name and object collection already exist, the utility terminates without

importing the tape(s). Using this option in the command line overrides the default.

- `-addAsInstanceIfNameExists`

  Attempt to add the tape object as an instance of an existing object in the Core database. The tape object must have the same object name and object collection, components, and checksums as the object in the database.

- `-useImportDateAsArchiveDate`

  Changes the imported object's original archive date to the date of import on the destination system. This doesn't change the original archive date in the exported XML file or in the original system where the object was exported from, only on the system where the object was imported.

5. In the DIVA Web App, navigate to the *Resources Management* > *Tapes*. Imported tapes can be left externalized, but to restore the objects on a tape it must be inserted into the library.
6. Click the Insertbutton at the bottom of the tape list to open the Insert Tape form.
7. Select the appropriate Robot Manager Name using the menu.

   An error is displayed if no Robot Manager is selected.
8. Select the appropriate CAP ID using the menu.

   An error is displayed if no CAP ID is selected.
9. Use the slide control to select the priority value for the insert operation, or enter the priority into the text box. If you check the Default Priority check box, the job uses the default priority set in the configuration.
10. Restoration of the objects on the imported tapes is possible after the tapes are inserted.

Example:

The tape with barcode number 000131 also contains objects that are spanned across the tape with a barcode of 000120. When tape 000131 is exported, the exported XML file is named Tapeset-000131.xml. This XML file also encompass the objects from tape 000120, and both tapes are ejected from the library.

After all objects from both tapes are exported to the XML file, all instances on each tape (and references to the tapes themselves) are removed from the Core database. The XML file is then copied to the `%DIVA_HOME%\Program\diva\bin` folder of the destination DIVA system where it is to be imported.

Import the metadata for this tape into the tape group MOVIES using the following command:

```
importtapes MOVIES Tapeset-000131.xml
```

After the tape's metadata successfully imports into the database (check the DIVA Web App Job History view), both tapes and their objects are considered externalized and can both be entered into the library with the Insert Tape command.

Import Tape Optional Behavior

The three main ways that a tape object can be treated during the import process are as follows:

- Imported as a new object:
  Normally, when a tape object is imported by the utility it is imported as a new object. This can only occur when the object name and object collection for the tape object doesn't exist in the target DIVAsystem. In the event of a naming conflict, the default behavior is to terminate the import operation without importing any tapes or objects.
  When new objects are imported into the target DIVA system, the import function only looks at the XML and FFM files and doesn't read directly from the tape structure. SPM is also automatically notified and if the object matches any of the SPM filters, then SPM initiates the required actions for the object.
- Skipped:
  A tape object can be skipped if the `-skipIfNameExists` flag is passed to the Import Utility. If there is another object already in the Core database that has the same object name and object collection as a tape object being imported, and the `-skipIfNameExists` flag is set, the object is skipped. The object instance on the tape is not recorded in the Core database (it is considered deleted by DIVA), and processing continues with the next tape object in the import metadata.

Caution: Be careful when skipping objects because the tape object that is skipped may or may not actually be the same as the object in the database. The tape object that had the naming conflict may in fact contain different content than the existing one in the Core database (content that should be preserved). If a tape is imported and then repacked, objects that were skipped won't be copied to the new tape and the old tape is reclaimed. If all objects on a tape are skipped (and the tape is made writable), the tape is marked for deletion and new objects overwrite existing objects on the tape. If you write new objects to the tape after the last object on tape is skipped, that tape instance is immediately overwritten.

- Added as an instance of an object already existing in the Core database:
  An object can be imported as an instance of another object if the `-addAsInstanceIfNameExists` flag is passed to the Import Utility. If there is another object already in the Core database that has the same object name and object collection as a tape object being imported, and the `-addAsInstanceIfNameExists` flag is passed, an Import as an Instance is attempted.
  First, the checksums for the tape object are compared to the checksums in the database object that matches it. If a match is produced (for each object component), the object is imported as an instance of the matching object. The comments, archived path root, archive date, uuid, storage plan, or tape group, of the imported object are lost and become that of the object already in the Core database.
  A new object instance ID is assigned every time the utility imports as an instance. If the checksum type of the object components in the database doesn't match

the checksum type in the imported object, or if one of the two objects has checksums that are missing, the tape object is not imported as an instance. This is considered a checksum mismatch and the import processing halts. However, if both the `-skipIfNameExists` flag and the `-addAsInstanceIfNameExists` flag are passed to the Import Utility (and a tape object matches one that already exists in the Core database), the utility first tries to import the object as an instance by comparing checksums. If this attempt fails the object is skipped and processing continues.

Note: SPM is not notified when importing as an instance. If the object matches any of the SPM Filters then SPM won't initiate the required actions for the object.

The Import Date as the Archive Date

The TapeImport command line utility provides an additional command line switch named `-useImportDateAsArchiveDate`.

Using this switch during object import causes the date of the imported object to be used as the date of the object archival in the system where it is being imported. The original archive date is not replaced in the XML export or on the original DIVA system, it is only replaced for the object on the imported system.

This feature supports tapes with spanned objects in the same way as regular tapes.

Error Conditions

If the tape media is not recognized by DIVA an error is generated specifying what occurred.

If the import process fails and DIVA detects a database error, the import process is terminated and any operations performed during the failed import is rolled back and not saved in the system.

In the case where the checksum comparison failed (or the checksum is not present) for one or several objects, the entire import process is stopped and the database transaction is rolled back.

If the `-skipIfNameExists` flag is used, the checksum verification still executes. However in this case an unverified (mismatched) object is skipped instead of stopping the entire import process.

All errors are displayed on the view and written to the log file. When using the `-skipIfNameExists` flag, the view messages and log file must be checked to determine whether all content intended to be imported was processed successfully. This option is not compatible with automated workflows since it may require operator intervention and decision.

Warnings and Limitations

Complex objects that are compared this way must have been archived in the same exact order to pass the checksum verification.

The Import Utility doesn't compare UUID, object ID, archive dates, or site ID values. The comments, archived path root, archive date, uuid, storage plan, or tape group, of the imported object aren't preserved when being added as an instance.

The utility doesn't enable the import of a set of tapes that contain an object with more than one instance on the tapes. An import metadata file having an object with more than one instance appearing within an exported tape set is not allowed. The export utility prevents this from happening.

Encrypted Tape Import

Specify only the destination tape group, and the folder containing the xml metadata file and the optional Keystore file to import encrypted tapes.

If a Keystore file is not present in the export folder, and you are attempting to import encrypted tapes, the tape encryption key for every tape must match the encryption key of the destination tape group.

If the Keystore file is present in the export folder, it is opened using the password in the DIVA Configuration of the new system, and therefore the passwords must match. If it doesn't match, the Keystore password is requested one time before failing the import. During the import the encryption keys are compared to the encryption key hash and salt to validate the keys. If any key is not valid, the import terminates.

Example: (DAMIEN CHECK THIS EXAMPLE, IT STILL SHOWS ORACLE)

```
Importer default D:\workspace\Core\bin\exported\2017-03-25--
10.37.14
================================================================
Core Tape Importer

Copyright (c) 1999, 2017 Oracle and/or its affiliates. All rights
reserved.
All rights reserved.
================================================================
The import completed successfully.
(Code 0)
See Tape Drive Encryption.
```

Cloned Tape Import

To restore the unidirectional Storage Link with the same Source and Clone Tape designations, the clone Storage Link must be imported first. Otherwise, the relationship becomes reversed; that is, the source tape becomes the Clone Tape and vice-versa. This is not critical because these tapes are clones of each other.

For example:

```
ImportTapes "GroupB"
"D:\diva\diva_std_730_jff\Program\Core\bin\exported\2019-04-04--
10.33.03\tapeset-3L2048.xml"
```

The import completed successfully. (Code 0)

Importing the associated tape requires the `-addAsInstanceIfNameExists` option, because the same objects are imported from the associated tape.

For example:

```
ImportTapes -addAsInstanceIfNameExists "GroupA"
"D:\diva\diva_std_730_jff\Program\Core\bin\exported\2019-04-04-
10.33.03\tapeset-3L2042.xml"

The following objects to be imported have the same Object Name and
Collection as objects already in the DIVA database... 1. A19/A 2.
A18/A 3. A17/A 4. A16/A 5. A7/A 6. A21/A 7. A20/A W A R N I N G ***
The 7 Virtual Object names listed above exist on the tape to be
imported, but ALSO exist in the DIVA system you are importing TO. *
You have chosen to first COMPARE the component CHECKSUMS of the
objects that have naming conflicts. If that match succeeds, these
objects will be imported as instances of the objects they match to.
* If the checksums match and you import as an instance, you may
lose the Comments, UUID, Archive Date and/or archived path root
that were stored in the import metadata. * If they don't match, the
import will fail, and then will exit...

The import completed successfully. (Code 0)
```

Bulk Tape Import

After inserting all tapes into the new system in the default tape group, the Tape Protection Setting for all tapes can be disabled. The setting on the Inserted Protected Tapes can be changed by clicking the three dots next to the tape barcode on the *Resources Management* > *Tapes* view and selecting Protect Tape from the menu.

Example Import

The tape with barcode number 000131 also contains objects that are spanned across the tape with a barcode of 000120. When tape 000131 is exported, its exported XML File is named Tapeset-000131.xml. This XML file also includes the objects from tape 000120, and both tapes 000131 and 000120 are ejected from the library. After all objects from both tapes are exported to the XML file, all instances on each tape and references to the tapes themselves are removed from the Core database.

The XML file is then copied to the `%DIVA_HOME%\Program\Core\bin` folder of the target DIVA system. The command `importtapes MOVIES Tapeset-000131.xml` results in the metadata for this tape being imported into the tape group MOVIES.

When the tape's metadata has been successfully imported to the database (check the DIVA Web App *Content Management* > *Jobs* queue, or Job History), both of the tapes and their objects are considered externalized and can then both be entered into the library with the Insert Tape command.

Importing of WORM Media is supported by DIVA. However, when importing DIVA WORM media into a release earlier than DIVA 7.4, DIVA ignores the WORM flag (set to false) and logs it in the DIVA log. The device is displayed in the DIVA Web App as a tape but not usable if finalized or no WORM drive is connected to the system.

Export and Import Tape Jobs

The Export Tape job type enables one or more tapes containing objects to be exported to another independent DIVA platform (for example, at a remote disaster recovery or partner site).

The metadata of each tape (that is, the object names and collections it contains and their location on the tape itself) are maintained in the Core Database. For complex objects this information is also in the Metadata Database. Additionally, the metadata of each tape is saved to an XML file when the tapes are exported. The XML file transfers the metadata of each tape into the other DIVA system's database when the tapes are imported.

The Export Tapes command is not used for transferring tapes between two or more Managed Storage controlled by the same Manager. Tapes (and the instances they contain) exported from DIVA using this command are also removed from the DIVA database. If the object being exported is the last (or only) instance of that object, it is removed entirely from the database.

The following parameters have been added for export and import tape functions. XML metadata files exported from all previous DIVA versions aren't supported.

During an export and import of WORM media, whether the media is Write-Once and the media is a cartridge is identified in the exported XML file. This information is imported with the attributes *isWriteOnce* and *isCartridge* being either true or false.

Importing WORM media is supported by DIVA. When importing DIVA WORM media into a DIVA version earlier than DIVA 7.4, DIVA ignores the WORM flag (it is set to false), and logs it in the Manager log. The device is visible in the DIVA Web App as a tape but unusable if finalized, or if no WORM drive is connected to the system.

The following table describes the export and import parameters:

Parameter	XML Element or Attribute	Notes
ObjectId	Attribute of object element	Not imported. A new object ID is generated during import.
uuid	Attribute of object element	Imported if present, otherwise a new UUID is generated.
numFolders	Attribute of object element	
format	Attribute of object element and attribute of the tape element	0 = legacy 1 = AXF -1 = unknown
numFolders	Attribute of object element	
isHeaderValid	Attribute of object element	

Parameter	XML Element or Attribute	Notes
isComplex	Attribute of object element	
footerBeginPos	Attribute of the element	If exists in database
footerEndPos	Attribute of the element	If exists in database
compOrderNumBegin	Attribute of the element	If exists in database
compOrderNumEnd	Attribute of the element	If exists in database
fileFolderMetadataInfo	Element	Valid for complex objects
fileFolderMetadataInfo-elem	Element	Valid for complex objects
checksums and checksum	Element	Not valid for complex objects

Exporting a Tape

To export one or more tapes, do the following:

1. Navigate to the *Resources Management* > *Tapes* view.
2. Click the three dots or select the check box next to the tape barcode.
3. Select *Export Tape* or click the Export Tape(s) button at the bottom of the tape list.
4. Fill in the required information in the in the form.
5. To execute the job, click *Submit*.

The following fields are included in the Eject Tape Job view:

- **Comments**

  Enter any desired comments in this field.

- **Delete from DB**

  When selected removes selected tape(s) from the Exported Tapes list.

- **Exported Tapes**

  This area displays tapes selected for export. These tapes and the instances they contain are removed from the DIVA Database after being exported. If required, select any tapes to be removed from the Exported Tapes list and click Remove Selected.

- **Priority**

  The priority level for this job. The level can be in the range 0 to 100, or the value Default. The value 0 is the lowest priority and 100 the highest priority. Move the

slide control to increase or decrease the job priority, or enter the desired priority number in the Priority Value text box.

There are six predefined values as follows:

- Min
- Low
- Normal
- High
- Max
- Default

If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

Export/Import Error Handling and Failure Scenarios

Export Failed Error Message

```
Robot Core Error: Error while ejecting tapes:
StatusCode[70:INTERNAL_ERROR]
```

```
Request step is STEP_WAITING_FOR_OPERATOR()
```

Resolution:

Check that the CAP where tapes are being ejected to has not reached its capacity. Even if the CAP is empty, if more tapes than the capacity of the CAP are exported a successful export operation can't be completed. This is specifically an issue with sets of spanned tapes and the number of tapes in that spanned set is greater than the number of tapes supported by the CAP. In this case, eject the tapes first then perform the export.

Invalid Parameter Error During Export

```
Invalid parameter: Tape Y00105 must be included into export list
```

Resolution:

When selecting the tapes for export, you can possibly see more tapes available in the tape window than initially selected. If a tape has objects that are spanned onto another tapes, these tapes are also included. In this case, select all of the spanned tapes from this list in order for the export to succeed.

Tape Already Exists Error During Import

```
The following errors were found in tapeset-J00026.xml\Tape J00026
already exists in DIVA. Consider performing a tape Insert
operation...
```

Resolution:

A tape with the same barcode as the one being imported already exists in the DIVA system. It is likely that the tape metadata for the tape you want to import already exists

in the Core database and you just need to perform an Insert Tape operation to use the tape. Verify the tape contains the correct objects by using the DIVA Web App on the *Content Management* > *Catalog Browsing* view.

Unsupported Type Error During Import

```
The following errors were found in tapeset-[Y00109].xml\Tape
Y00109 has unsupported type 19.
```

Resolution:

The type in the message refers to the `mediaTypeId`. The `mediaTypeId` is an ID that represents the type of tape media being exported. DIVA exports a mediaTypeId field that corresponds to the Type column on the *Resources Management* > *Media* view in the DIVA Web App. A Synchronize DB call may be required to update the `mediaTypeId` and/or update the hardware to be compatible with a newly imported tape. Ensure that the block size and total size of the *mediaType* in the source DIVA system matches the *mediaType* definition in the destination.

Import Process Terminated Without Importing

There are several reasons why the import process may terminate without completing successfully including the following:

- When using complex objects, the FFM files must be in the same folder as the XML files for importing. If the FFM file is not found, the import process will terminate and an error will be written to the log file.
- If the object name and object collection already exist, and the -skipIfNameExists or -addAsInstanceIfNameExists options aren't passed, the utility will terminate without importing.
- If DIVA detects a database error the import process will be terminated and any operations performed during the failed import will be rolled back and not saved in the system.

Export/Import Through the Java API

You can import and export tapes through the Java API, and also in the Java Initiator (delivered with the API). The following is sample output from the export and import of a single encrypted tape using the Java Initiator.

```
DIVA JInitiator - Using JavaAPI Version: 8.3 SNAPSHOT

0 = Exit
1 = Connect
2 = Archive Virtual Object
3 = Copy to New Virtual Object
4 = Copy to Group (new instance)
5 = Associative Copy
6 = Delete Virtual Object
7 = Delete Instance
```

```
8 = Delete File
9 = Delete File For Collection Mask
10 = Restore Virtual Object
11 = Restore Instance
12 = N - Restore
13 = Partial Restore
14 = TranscodeArchive
15 = Transfer
16 = Insert tape
17 = Eject tape
18 = Cancel Request
19 = Change Priority
20 = Get Request Information
21 = Get Partial Restore Request Information
22 = Get Finished Request List
23 = Get Virtual Object Info
24 = Get Tape Info
25 = Get Virtual Object Details List
26 = Get Files and Folder Names for Virtual Object
27 = Get Array List
28 = Get Server List
29 = Get Tape Group List
30 = Add Tape Group
31 = Delete Tape Group
32 = Require Instance
33 = Release Instance
34 = Get Storage Plan Names List
35 = Get Virtual Object List By File Name
36 = Get Archive System Information
37 = Lock Virtual Object
38 = Unlock Virtual Object
39 = Link Virtual Objects
40 = Enable Automatic Repack
41 = Export Tapes
42 = Import Tapes
100= close connection

Enter a command number : 41
*** exportTapes ***
Specify the Tape barcodes.
Barcode 1 <1S0009>: 3L2247
Add another? <N>:
Comment <>:
Set delete from database (Y/N) <Y>:
Priority <-1>:

--- Export Tapes ---
Tapes
3L2247
Comment:
Delete from database: true
Priority: -1
```

```
Submit[S] or Submit and Wait[W] <S>:
Status: Running
Request ID: 20272
Success
Press Enter to continue.
Request 20272 has completed successfully
DIVA JInitiator - Using JavaAPI Version: 8.3 SNAPSHOT
0 = Exit
1 = Connect
2 = Archive Virtual Object
3 = Copy to New Virtual Object
4 = Copy to Group (new instance)
5 = Associative Copy
6 = Delete Virtual Object
7 = Delete Instance
8 = Delete File
9 = Delete File For Collection Mask
10 = Restore Virtual Object
11 = Restore Instance
12 = N - Restore
13 = Partial Restore
14 = TranscodeArchive
15 = Transfer
16 = Insert tape
17 = Eject tape
18 = Cancel Request
19 = Change Priority
20 = Get Request Information
21 = Get Partial Restore Request Information
22 = Get Finished Request List
23 = Get Virtual Object Info
24 = Get Tape Info
25 = Get Virtual Object Details List
26 = Get Files and Folder Names for Virtual Object
27 = Get Array List
28 = Get Server List
29 = Get Tape Group List
30 = Add Tape Group
31 = Delete Tape Group
32 = Require Instance
33 = Release Instance
34 = Get Storage Plan Names List
35 = Get Virtual Object List By File Name
36 = Get Archive System Information
37 = Lock Virtual Object
38 = Unlock Virtual Object
39 = Link Virtual Objects
40 = Enable Automatic Repack
41 = Export Tapes
42 = Import Tapes
100= close connection
```

```
Enter a command number: 42
*** importTapes ***
Specify the file or directories to import
File 1 <>: D:\workspace\diva\bin\exported\2017-04-04--17.28.57
Add another file? <N>:
Group name <>: default
Skip Virtual Object import if already exists in database (Y/N) <N>:
Add as instance if the Virtual Object already exists in database
(Y/N) <N>:
--- Import Tapes ---
Files and Directories
D:\workspace\diva\bin\exported\2017-04-04--17.28.57
Group name: default
Continue? <Y>:
--------- Response ----------
Success
Press Enter to continue.
```

Adding a Tape Library, Corresponding Drives and Tapes into DIVA

Prerequisites

Software

- DIVA Server installed and accessible
- iSCSI initiator configured on the DIVA Server hosting the Robot Manager module

Hardware

- Remote Tape Library started
- Access to the network hosting the DIVA Server and Tape Library

Credentials

- Administrative access to DIVA Server

Environment Setup

Verifying the Environment

1. Ensure all required devices are powered on and connected to the same network.
2. Confirm that your administrative credentials allow full access to the DIVA software.

Confirm a Robot Manager

3. Open the Services panel on the DIVA server.
4. Confirm you already have a Robot Manager service. If yes, identify the port on which that Robot Manager is started. You'll need to create a new Robot Manager service using a distinct IP port number.

Setting Up the Remote Library

These are the main steps for adding a tape library to DIVA:

Ensure compatibility. Verify that your tape library is supported by DIVA.

Connect the library. Physically connect the tape library to your DIVA system according to manufacturer instructions.

Configure the library. Configure the tape library with the DIVA Web App.

Modify configuration files. If necessary, modify the configuration files to include the tape library in the DIVA system.

Test the connection. Ensure that DIVA communicates with the tape library.

Monitor and Manage. Monitor and manage the tape library through the DIVA Web App.

Identify the connection interface to the remote Library—either IP or iSCSI—depending on your hardware. Also, identify the models of Tape Drives and Tapes. You need these parameters to tune DIVA configuration files.

Connecting DIVA to a Remote Library

Connect DIVA to a remote library on the DIVA server where the new DIVA Robot Manager is configured and started.

Configuring Devices in Windows

1. Open Device Manager on the DIVA server.
2. Verify that LTO drives and an unknown medium changer are listed.
3. To prevent conflicts, don't install drivers for the medium changer.

Configuring DIVA Robot Manager

Editing Configuration Files

1. Navigate to `C:\Diva\Program\RobotManager\Conf`.

2. Make a backup copy of these three files, in case of editing errors:
   - `robotmanager.conf`
   - `drivetypes`
   - `tapetypes`
3. Edit the following files using a text editor:
   - `robotmanager.conf`:

     Update `SerialNumber` to match the serial number of your Library.

     Assign `ServiceName` (for example `<MY LIBRARY VENDOR NAME>_LIB`).

     Update `RM_PORT` to an IP port not assigned nor used on your server.

     Update `RM_ACS` to an ACS number not assigned nor used in any other Robot Manager on your DIVA Server.

     Remove the # from the line "`#RM_SCSI_ENABLE_MEDIA_TYPE_DETEC-TION_LAYOUT=LLLLLLTT`" This will report six Character Bar Codes i.e `TEST01` instead of `TEST01L6`

     If you already have an existing `robotmanager.conf` file in the directory, save your new file with a different unique name.
   - `drivetypes`:

     Select the appropriate tape type, by removing the # preceding the type you need.

     Example "#59 0x4C 0x3D HP-LTO6 R40 R41 45 46 48 49" will give you HP LTO6 Drive Types.
   - `tapetypes`:

     Ensure the correct configuration for LTO6 tapes.

Installing Robot Manager Service

4. Open a command prompt in the `RobotManager\Bin` directory.
5. Install the service using the command:

   ```
   robotmanager.exe -install <MY LIBRARY VENDOR NAME>_LIB
   ```
6. Start the service and verify its status in the *Services* panel.

Syncing and Adding the Library in DIVA

Adding the Robot

1. In the DIVA Web App, browse to *Library Storage* > *Robots*.
2. Add a new robot with the following data:
   - *Name:* `<MY LIBRARY VENDOR NAME>_LIB`
   - *Address:* IP address of the DIVA Server running the Robot Manager service
   - *Port:* value set in `RM_PORT`

3. To update the manager, click *Save*.

Configuring Tapes and Drives

4. Synchronize the robot to detect drives and tape types.
5. Assign tapes to groups based on their intended use.
6. Set drives online in the DIVA Web App.

Final Configurations and Testing

Adjusting iSCSI Settings

1. Update `MaxTransferLength` in the registry to 80000 (hexadecimal):
   Path: `HKLM\System\CurrentControlSet\Control\Class\6E97B...`
2. To apply changes, reboot the DIVA server.

Testing the Configuration

3. Submit a test archive job in the DIVA WebApp.
4. Verify the tape group, format, and data-transfer operations.
5. Confirm data is successfully written and retrievable from the VTL.

Troubleshooting Tips

Common Errors and Fixes

Robot Manager Fails to Start. Check for mismatched serial numbers or incorrect port settings in `robotmanager.conf`.

Synchronization Issues. Ensure tape and drive types match in the configuration files.

Transfer Errors. Verify `MaxTransferLength` and ensure iSCSI is configured properly.

Logs and Diagnostics

- For detailed error messages, check logs in `C:\Diva\Program\RobotManager\Logs`.
- To identify mis-configurations, use the DIVA WebApp.

FAQs and Additional Resources

1. Can I use multiple robots in DIVA?
   Yes, assign unique ports and configure each robot individually.

2. What if the remote Tape Library is unavailable?

   Check SCSI configuration and connections.

Damaged Tape Utility Recovery

```
%DIVA_HOME%\Program\Utilities\bin\rdtu.bat
```

The Recover Damaged Tape Utility (RDTU) recovers object instances that reside on a damaged tape. The utility can recover instances that have valid copies on other available media (that is, internal tape or a connected disk array) within a local or remote DIVA system. There are no command line parameters. The settings and configurations are defined in the `rdtu-conf.xml` configuration file.

Tape Scanning for Indexing Objects

The tape scan functionality allows a DIVA user to inspect the content of LTFS formatted tapes, and to create new DIVA objects based on what is discovered. This is useful for importing a foreign LTFS tape and using its content.

This feature is implemented in the DIVA Manager module, and does the following for a given tape:

- Mounts a selected tape into a compatible drive
- Scans the content of the tape in order to add new objects or new instances of existing objects

  The new objects or instances are available for other operations such as copy, restore, and metadata extraction.

See *Supported Tape Formats* in the *DIVA Supported Environments Guide*.

Tape Scan Job Parameters

You can initiate tape scan jobs from the DIVA REST API and the DIVA Web App. All the parameters are the same. The parameters are described in the following table:

Parameter	Description
Trial Mode	Scans a tape without making changes to the database. Previews what would be added to the object catalog before running a full tape scan.
Target group	When scanning an empty tape, this is the target group of the tape if it contains files to be indexed. The format of the target group must be RAW_LTFS.
Collection	Includes all the Objects created during the tape scan.
Comments	The comments associated with each new object created during a tape scan.

Parameter	Description
Included Files	(Optional) You can set this to a regular expression. All the files matching the pattern are indexed. The others are excluded. When this option is empty, all the files are included.
Excluded Files	(Optional) You can set this to a regular expression. All the files matching the pattern are excluded. This pattern is evaluated after *Included files*. When this option is empty, this filter is inactive.
Minimum file size	Files smaller than this customizable file size are excluded by the scan.

Submitting an LTFS Scan from the DIVA Web App

The Scan Tape action for tapes is available from the *Tape Action* menu. It is only available for tapes that report they are scannable. Scannable tapes are not in use, and either are not in a tape group, or are in a tape group of type Raw LTFS.

It is possible to scan up to 10 tapes at once with the same settings.

To submit an LTFS scan from the DIVA Web App, do the following:

1. Browse to *Resource Management* > *Tapes*.

   DIVA opens the *Tapes* page.
2. Check the boxes for the tapes desired.
3. Select *Action* > *Scan Tape*.

   DIVA opens the *Tape Scanning* dialog.

   The *Target Group* drop-down menu is populated with tape groups that have the format Raw LTFS set.
4. Enter values into the fields, as desired.

Note: Telestream recommends that you first do a tape scan in *Trial* mode. This way you can confirm the scan will deliver the expected results. Then, to import the objects into DIVA, redo the scan without *Trial* mode.

5. Click *Submit*.

   DIVA creates a Scan Tape job for each tape selected.

Viewing Scan Tape Jobs

To view Scan Tape jobs, do the following:

1. Browse to *Content Management* > *Jobs*, or *Content Management* > *Job History*.
2. Click the Filters button.

   DIVA opens the Filters dialog.

3. From the *Type* drop-down menu, select *Scan Tape*.
4. Click *Apply Filters*.

   DIVA displays a list of Scan Tape jobs.

See also *Displaying Object Properties*.

Re-indexing Existing Objects

During a tape scan, DIVA may have to re-index objects that already exist in the database. Possible scenarios are as follows:

- The object already exists because the same tape was scanned multiple times. DIVA doesn't create new objects for files that already are associated with an existing object. The verification is made using the name of the file and its position on tape. In this case, DIVA skips the file and logs an event.
- The object already exists because a file with the same name already has been indexed, and a tape scan job is trying to create a new object with the same name and same collection. This may happen if two files with the same name are on two different directories or on two different LTFS tapes. When this occurs, the tape scan job adds a suffix to the name of the object so that the name remains unique. The suffix is a space followed by a number between parentheses. The number is incremented until the object name is unique.
  Examples:
  - MyFile.mxf
  - MyFile.mxf (1)
  - MyFile.mxf (2)

An object with the same name and same collection already has been archived in the past. The tape scan job also adds a suffix to the name of the object, as described in the previous scenario.

Checksum and Content Verification Support

This chapter provides information about using checksum and content verification.

Topics

- About Checksum
- Checksum Support Configuration
- AXF and TEXT Genuine Checksum Modes

About Checksum

The Checksum Support and Content Verification program provides the additional levels of verification checksum needs to generate and verify files managed by DIVA. Checksum algorithms in DIVA include MD5, SH1, MDC2, RIPEMD160.

Note: Telestream recommends the default MD5 checksum algorithm. DIVA retains backward-compatibility with the other checksum algorithms, but MD5 and SHA-1 deliver the best results.

During the archive process, the actor generates checksums automatically, and stores them in the database. You can verify these checksums when genuine checksums are provided before archive. Otherwise, DIVA doesn't verify these checksums until DIVA performs an initial read-back or restore operation.

DIVA verifies checksums while transferring data from a server, or while reading data from a source server or storage medium.

DIVA verifies checksums while retrieving an object from a storage medium for the following types of jobs:

- Restore
- Copy
- Repack
- Transcode

During read-backs from storage (Verify-Following-Write), or from the source server (Verify-Following-Restore), DIVA doesn't verify checksums for Partial File Restore jobs.

When an object contains multiple files, DIVA generates a checksum file, and verifies it later for each of the component elements. The following types of checksum sources are distinguishable:

- Genuine Checksum
- Archive Checksum
- Deferred Checksum

While archiving files in the TEXT Genuine Checksum mode, DIVA compares the checksum values of the files against values stored in an external checksum file. DIVA archives non-matching files with checksums generated by DIVA, without external checksum files.

Note: The TEXT Genuine Checksum is a customer-specific implementation which supports only MD5. Unicode is not supported, and checksums must be in an MD5 text file.

Viewing Checksum Values

To view the checksum verifications and failures through the DIVA Web App, do the following:

1. Browse to *Content Management* > *Catalog Browsing* view.
2. View the *Checksum Status* column.

Checksum Support Configuration

Configure the Checksum Support functions through the DIVA Web App using the Sysadmin account. The following sections describe the settings for each option.

Global Checksum Parameters

Each of the global parameters affects all Checksum Support settings throughout the system.

Enabling the Checksum Feature

To enable the checksum feature, do the following:

1. Launch the DIVA Web App, and log in with the sysadmin account.
2. Scroll down to *Configuration / Resources / General Settings /Checksums*.
3. Do one of the following:

- To enable checksum, slide the selector for *Manager: Checksum Feature Is Enabled* to the right. This is the default setting.
- To disable checksum, slide the selection for *Manager: Checksum Feature Is Enabled* to the left.
- To save your checksum configuration, click *Save All Changes*.

Manager: Default Checksum Type

There are several checksum algorithms supported by the system including MD5, SH1, MDC2, RIPEMD160. DIVA uses MD5 as the default checksum.

Each checksum type is associated with an ID Number. you use the menu list to change the default type and select the type of checksum desired.

Manager: Number of Retries Following Failed Checksum

This parameter sets the number of times the system retries the operation after a failed checksum. The default setting is one retry. Telestream recommends leaving this setting at the default value.

To set the number of retries, enter the number of retries allowable for your data and system into the *Manager: Number of Retries Following Failed Checksum* field.

Manager: Select Different Drive Per Retry on Failed Checksum

This parameter determines whether, after a failure, DIVA retries the checksum on the same drive or on a different drive.

To set this option, do one of the following:

- To have DIVA retry on the same drive, deselect the checkbox. This is the default setting.
- To have DIVA retry on a different drive, select the checkbox.

Adjusting Checksum Support for Unmanaged Storage

To adjust the Checksum Support configuration for Unmanaged Storage, do the following:

1. Browse to *Configuration* > *Resources* > *System Settings* > *Unmanaged Storage*.
2. To display the checksum support configuration, edit the configuration for the desired Unmanaged Storage, then go to the *Advanced Settings* tab.

The following list describes the options available:

External Checksum Source

For the system to read the checksum from the external source server, enable the selection *For External Checksum Source*. This initiates an immediate checksum

calculation to compare the checksums and verify the initial transfer. Selecting *No* disables Genuine Checksum support from external source servers.

Checksum Type

Select the checksum type desired for the first verification from the menu. The checksum type and GC mode must match the settings implemented at the source server.

DIVA uses the Genuine Checksum for the first verification. After the first verification, DIVA uses the default type.

GC Mode

Select the Genuine Checksum Mode from the menu. This notifies the actor of the format of the files that contain the checksum data.

Verification Following Archive (VFA)

When Verify Following Archive (VFA) is turned on (slider to the right), the initial transfer from the source server results in a read-back operation. Therefore, the data is read twice for verification. After reading the data twice, DIVA compares the two checksums. If the checksums match, the verification is completed successfully. If they don't, then verification has failed.

Note: To activate the VFA check box, turn off Genuine Checksum. If Genuine Checksum is turned on, the VFA check box is grayed out and isn't selectable.

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.

Following Restore (VFR) Verification

When Verify Following Restore (VFR) is turned on (slider to the right), the final transfer to the destination server results in a read-back operation. Therefore, the data is read twice for verification. After reading the data twice, DIVA compares the two checksums. If the checksums match, the verification is completed successfully. If they don't, then verification has failed.

Note: The Genuine Checksum setting has no bearing on the VFR setting.

Verify Following Restore is not compatible with Complex Objects or the -AXF option. Verify Following Restore reads back the restored content from a video server to confirm that it is not corrupt. Using the -AXF option doesn't create a checksum-verifiable restore. It creates an Object export in an AXF wrapper. The VFR and -AXF options are mutually exclusive and should not be part of the same workflow.

Configuring Checksum Support for Arrays and Disks

Checksum Support for Arrays and Disks is configured through the DIVA Web App Disks view. 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 DIVA. When turned ON, the system performs a read-back of what was just written and compare the checksums for verification.

The VW column in both the Arrays area and Disks area indicates whether the Verify Write function is on or off for the particular array and disk. The default setting is *OFF*.

If there is nothing defined in the VW column on the Disk area the system uses the setting defined in the Array VW column.

To override the setting defined in the Array VW column for a specific disk, do the following:

1. In the *Disks* area, select the disk requiring configuration.
2. Click *Edit*.

   DIVA displays the *Edit Disks Entry* dialog.
3. From the *Verify Write* menu, select *ON*, *OFF*, or *Inherit From Array*.

   If you select *Inherit From Array*, Verify Write uses the setting identified in the array for this particular disk.

   The selection you make in the *Edit Disk Entry* dialog box is shown in the Disks VW column.

Configuring Checksum Support for Tape Groups

Verify Write (VW) for tape groups also is configurable. DIVA displays the VW column in the DIVA Web App *Tape Groups* view. This is the only place where configuration of Verify Write is available for the tape groups.

To configure a tape group, do the following:

1. Select the tape group requiring configuration.
2. Click *Edit*.
3. From the *Verify Write* menu, and select *ON* or *OFF*.

   The selection you make is displayed in the *Tape Group* VW column.

When DIVA writes a file to a particular tape group, the setting for that tape group is applied to the file. The default setting for tape groups is *OFF*.

Configuring Checksum Support for Actors

This setting determines whether DIVA automatically selects the actor for the Verify Tape workflow. By default, all actors are included, but you can exclude actors as needed.

Similar to the configuration for disks and tape groups, Verify Tape for Actors is configurable. To configure Verify Tape for Actors, do the following:

1. Browse to *Configuration* > *Resources* > *System Settings* > *Actors* and select the actor requiring the configuration change.
2. Click *Edit*.
3. Open the *Advanced Settings* > *Jobs* menu.
4. Do one of the following:

- To include an actor, slide the *Verify Tape* slider right.
- To exclude an actor, slide the *Verify Tape* slider left.

AXF and TEXT Genuine Checksum Modes

There are two additional Genuine Checksum modes as follows:

AXF Genuine Checksum Mode

This mode enables DIVA to archive all files and sub-folders in a specified AXF file while comparing their checksum values against known values stored in the AXF file. This workflow is typically combined with a Restore job with -AXF in the Job Options.

Requirements

The AXF Virtual Object containing the files to be archived must contain checksum information for each file. The supplied checksum in the AXF Virtual Object must be the expected type as defined in the configuration.

AXF Genuine Checksum Mode

There are specific requirements and limitations when using the AXF Genuine Checksum Mode as follows:

- The AXF file containing the files to be archived must contain checksum information for each file.
- The checksums must be the expected type as specified in the configuration.
- This workflow only works with AXF jobs generated by DIVA.
- 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. The -AXF option exports an Object in an AXF wrapper, not a real restore. These options are mutually exclusive and shouldn't be part of the same workflow.

Configuring DIVA Web App Settings

To configure AXF Genuine Checksum Mode in the DIVA Web App, do the following:

1. Create a new server entry with the source server Type set to either DISK, FTP_STANDARD, or EXPEDAT as appropriate.

   If you are required to specify an appropriate Root Path, this path along with the input files specified during the archive job is used in determining the location of the checksum file.

   For example, if the source server Type is DISK, you can set the Root Path to `D:\root`. If the source server Type is FTP_STANDARD, you can set the Root Path to `/root`.
2. Set the External Checksum source server to *YES*.
3. Set the checksum type to the expected checksum type (for example, MD5).
4. Set the GC Mode to *AXF*.
5. Click *SAVE*.
6. To notify the Manager of the configuration, select *Notify Manager* from the menu.

Archiving an Object with Genuine Checksum

To archive an object with Genuine Checksum through an AXF transfer, do the following:

1. Browse to *Content Management* > *Archive Submission*.
2. Fill in the required information in the text boxes.
3. From the source server menu, select the server entry that was created in the Configuration procedure.
4. Enter the desired File Path Root.
5. Enter the remaining parameters in the archive submission form and click *Submit*.

Limitations

The workflow described only works with AXF jobs generated by DIVA.

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. The -AXF option doesn't create an actual restore, but rather an object export in an AXF Wrapper. These options are mutually exclusive and should not be part of the same workflow.

TEXT Genuine Checksum Mode

This mode enables DIVA to archive all files and sub-folders in a specified folder while comparing their checksum values against known values stored in an external checksum file.

TEXT Genuine Checksum Mode

There are specific requirements and limitations when using the TEXT Genuine Checksum Mode as follows:

- A checksum file must be present in the folder specified by the Root File Path.

- Checksum files must end with an MD5 file extension.
- The checksum file name (excluding the extension) must be associated with the folder name that contains all files to be archived. This folder must exist.

  For example, if the checksum file is `D:\Data\Video\NewTitle.md5`, then all files located in the `D:\Data\Video\NewTitle` folder 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.
- Absolute path names are supported on Windows to a maximum of 4000 characters. Relative path names are limited to 256 characters on Windows systems.
- 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.

Configuring DIVA Web App Settings

To configure TEXT Genuine Checksum Mode in the DIVA Web App, do the following:

1. Create a new server entry with source server 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 job is used in determining the location of the checksum file.

   For example, if the source server type is DISK, you can set the Root Path to `D:\Data`. If the source server type is FTP_STANDARD, you can set the Root Path to `/Data`.

   For further details, see The Root File Path.
3. Set *External Checksum Source Server* to *YES*.
4. Set the checksum type to *MD5*.
5. Set the GC Mode to *TEXT*.
6. Click *SAVE*.
7. Notify the Manager of the configuration by selecting *Tools* > *Notify Manager* from the menu.

The Root File Path

The Root File Path must point to the folder containing the checksum file. Therefore, set the correct file and folder paths in the *Server and Archive Job* form. For example, if the checksum file is located in `D:\Data\Video\NewTitle.md5` (or `/Data/Video/NewTitle.md5` for FTP type), you set the appropriate file and folder paths as follows:

Server (Root Path)	Archive Job (File Path Root)	Archive Job (Files)
`D:\`	`Data\Video\NewTitle`	`*`
`D:\Data`	`Video\NewTitle`	`*`
`D:\`	`Data\`	`Video\NewTitle\*`
`D:\`		`Data\Video\NewTitle\*`

Server (Root Path)	Archive Job (File Path Root)	Archive Job (Files)
`/`	`Data/Video/NewTitle`	`*`
`/Data`	`Video/NewTitle`	`*`
`/`	`Data/`	`Video/NewTitle/*`
`/`		`Data/Video/NewTitle/*`

Archiving Objects with Checksum

To archive objects using checksum verification through the DIVA Web App, do the following:

1. In the DIVA Web App, browse to *DIVA* > *Archive Submission*.
2. From the Source drop-down list, select the server entry that was created in the configuration stage.
3. Enter the desired File Path Root in the File Path Root text field.
4. Enter the path to the location of the checksum files in the Files field and append your entry with a wildcard symbol (an asterisk).
5. Enter *-r* In the Options field.
6. Enter the remaining parameters in the job form and click *Submit*.

Limitations

The following limitations apply when using checksum verification.

- DIVA can't open or create files on a Windows file system if the full path exceeds 256 characters. The Root Path must have no more than a total of 256 characters.
- Only ASCII, non-UTF-8 encoded checksum files are supported.
- Each line in the checksum file must begin with an MD5 checksum, followed by 2 spaces, and then the full file P\path to the referenced file.

DIVA Content Migration Tool

This chapter provides information about the use of the DIVA migration tool.

Topics

- About DIVA Migration Tool
- Migration Tool Installation
- Configuration
- Starting and Stopping the Service
- Migration Tool
- Migration Task Functions and Parameters
- Advanced Migration Jobs

About DIVA Migration Tool

The DIVA Migration Tool is an embedded Windows service. With the DIVA Migration Tool, you can schedule and run jobs that migrate media between different storage systems in DIVA.

Migration jobs are initiated in DIVA connected to the Manager. Migration jobs are stored in the Migration Service database. The Migration Tool starts migration tasks, monitors the status of jobs controlled by the Manager, and updates the status of existing migration tasks in the Migration Tool database so that the DIVA Web App displays the status to users.

Migration Tool Installation

The Migration Tool is installed, as part of the standard DIVA installation, in the `%DIVA_HOME%\Program\` folder. You can install the Migration Tool on the Manager server or any other server capable of communicating with the Manager via TCP/IP. Confirm connectivity by pinging the Manager server from the server on which you plan to install the Migration Tool.

Migration Tool licensing

Migration Service retrieves the active license from the DATA service on startup. If the feature is not enabled in the license, all Migration Service endpoints respond with HTTP 403.

The DIVA Web App displays the Migrations tab only if the Migration Tool is enabled in the DIVA license.

Windows Files and Folders

These files and folders are used with the Migration Tool and may be edited for configuration purposes after completing initial DIVA installation:

- `%DIVA_HOME%\Program\MigrationService\bin\migration_service.bat`
- `%DIVA_HOME%\Program\conf\migration_service\log4j2.xml`
- `%DIVA_HOME%\Program\conf\migration_service\application.xml`
- `%DIVA_HOME%\Program\log\migration_service\MigrationService.log`

Configuration

The Migration Service requires a valid configuration file during install and startup procedures. The default Migration Tool configuration file is named `application.yml` and is located in the `%DIVA_HOME%\Program\conf\migration_service\` folder.

There are no changes needed in the `%DIVA_HOME%\Program\conf\migration_service\application.yml` configuration file if the Migration Service is started on the same server as the DIVA Manager.

Logging Settings

The Migration Tool logs are located in `%DIVA_HOME%\Program\log\migration_service\`. The Migration Tool logs are automatically archived and divided into separate files each time the current log file reaches its size limit. Set the following Migration Tool logging parameters in the `log4j2.xml` file.

To ease tracing progress of TAPE Migration in Operation as well as troubleshooting, Manager logs all events and tasks related to migration of a TAPE via CopyTape request to a dedicated file:
`C:\DIVA\Program\log\manager\diva_manager\manager.copy-tape-trace.log`

The `<Loggers>` element can be edited to modify the log levels. For example:

```
<!-- Migration Service -->
```

```
<Logger name="com.telestream.diva.migrationservice" level="debug"
additivity="false">
<AppenderRef ref="DIVAFileAppender" />
</Logger>
```

where the level can be set to `INFORMATION` or `DEBUG`.

Note: To ease tracing progress of TAPE Migration in Operation as well as troubleshooting, Manager logs all events and tasks related to migration of a TAPE via `CopyTape` request to a dedicated file: `C:\DIVA\Program\log\manager\diva_manager\manager.copy-tape-trace.log`

Starting and Stopping the Service

You can start and stop the Migration Tool from the command line or the Windows Service Management Console. The Windows Service short name displayed in the Windows Service Management Console is `DIVACoreMigrationService`, and the full name displayed is DIVA Core Migration Service.

To install the Migration Tool, open a command-line interface, and navigate to the folder where the `migration_service.bat` file is located. Run the batch file with the appropriate parameters required to perform the desired tasks.

To run the service on Windows, do the following:

1. Open a Windows command-line.
2. Change to the `%DIVA_HOME%\Program\MigrationService\bin` folder.
3. Execute a command similar to *migration_service.bat install*. This specific command installs the service.
4. To start the service, run `migration_service.bat start`.
5. Verify that the service starts and runs.

The migration_service.bat command-line syntax is `migration_service.bat {command} [option]`. Commands and options:

- *install*–Installs the module as a Windows service.
- *uninstall*–Removes the Windows service module.
- *start*–Starts the service.
- *stop*–Stops the service.
- *help*–Displays command and option help information, and then exits.

Migration Tool

DIVA has a Migrations menu and Migration Wizard.

Use the Migration Wizard to create new migration tasks. View the status and results of completed and currently running migration tasks and underlying migration jobs on the Migrations menu. You can pause, stop, resume, delete, cancel migration tasks using commands in the Migrations menu.

See *Migrations Button*.

Migration Task Events

For any selected Migration Task, the Migration Service includes migration task events. To view any selected Migration Task events, browse to *Migration Tasks* > *Migration Task Properties* > *Migration Task Events*.

DIVA displays job history on the *Content Management* > *Jobs* view and the *Content Management* > *Job History* view. DIVA loads events in descending order according to time and event id (by default). The *Job History* table highlights events with different colors based on severity. Red indicates an error, yellow indicates a warning, and white indicates informational messages. To refresh the entire view, click the *Refresh* button.

Migrating Sources to Targets (Administrators Only)

The Migration Wizard is a wizard with a series of dialog boxes. Each dialog box presents a set of choices for migration task parameters. DIVA displays succeeding dialog boxes based on your selections, until you configure all required parameters.

The *Start New Migration Task* menu item is located under the Migrations menu item.

The Migration Tool Wizard is only available to users with Administrator privileges.

To perform a migration using the wizard, do the following:

1. Browse to *Migrations* > *Start New Migration Task* in the DIVA Web App.
2. Select the Migration Type.
3. Select the Migration Source Type.
4. Use the Migration Source to select one or multiple source(s) and click *Next*.
5. Use the Migration Target to select one or multiple targets and click *Next*.
6. Enter the Migration Settings and click *Next*.
7. The Migration Summary view displays a summary of the selections made during the process. On this view either click *Create Migration* to schedule the job, click one of the numbered items to go back and make any necessary changes, or click a different menu item to cancel the job (there is no Cancel button on this view).
8. The final wizard view provides a unique Name for the Migration Task, and an optional Note to fill in migration description. To create this new Migration Task, click *Confirm*.

Migration Tasks View

The *MIGRATION TASKS* view is located in theDIVA Web App and displays information about migration tasks in a table containing the following columns. Some columns are filterable as noted.

- Actions (three dots allow Pause, Stop and Cancel job options)
- ID (this value is a number that uniquely identifies a Migration Task)
- Name
- Description
- Status (filterable)
- Type (filterable and the value can be Copy, Move, or Copy to Source)
- Progress
- Source Type
- Submitted Time
- Start Time
- End Time

Click the *Migration Task ID* to display the *MIGRATION TASK PROPERTIES* view. The properties view shows detailed information and the status for each source affected by the migration Job. The *Migration Task Jobs* table in the *Migration Task Properties* view displays 1000 objects at a time and supports pagination.

The Migration Tasks Events table displays all events associated with the selected Migration Job.

See *Migrations Button*.

Migration Task Functions and Parameters

This sections describes the migration task functions and parameters.

Caution: The Migration Tool doesn't allow changes to migration task parameters or options after a migration task is submitted. If you want to change an existing migration task, delete the existing task, and create a new migration task with the parameters and options desired.

Migration Task Properties

You create Migration tasks using *Migrations* > *Start New Migration Task*. Configure each of these mandatory parameters for each migration you perform:

- **Migration Type**—Specifies the type of migration. Select from Copy or Move.
- **Migration Source Type**—Identifies the source type. Select from Tapes, Archived Objects, or Import file (from local machine).

- **Migration Source**—Describes the set of instances that must be migrated.
- **Migration Target**—Specifies the location where migrated instance are to be placed.
- **Migration Settings**—Specifies various additional parameters and options available for the migration job.
- **Migration Summary**—Summary of all selected and configured options to confirm before submitting the migration job. Click the appropriate item to submit, modify, or cancel the migration job.

Migration Tasks Actions

These actions can be performed on migration tasks:

- **Start**—Starts the migration task which is in pending status.
- **Pause**—Saves the current state of the migration and pauses task execution. After resuming, the task continues execution from the point where it was paused without recollecting data.
- **Delete**—Deletes the task from the database.
- **Resume**—Resumes the task execution for a job in a Paused status.

Migration Task Status

Based on the migration task life cycle and possible user actions, each migration task holds its state in the Migration Task status; for example, Pendng, Completed, and Aborted.

Migration Task Parameters

This section describes (in detail) each of the migration task parameters.

Migration Type

The Migration Type defines the type of migration being performed. DIVA supports the following Migration Type options:

- Copy Objects

  This strategy migrates the configured number of instances from the Migration Source to the Migration Target, but does not remove original instances from the Migration Source.

- Move

  This strategy migrates the configured number of instances from the Migration Source to the Migration Target, and removes original instances from the Migration Source.

Migration Source Type

This migration source type defines the set of instances for migration. The migration source can be provided using list of instances; for example TAPE bar-codes, Object Name and Collection.

Valid values for the Migration Source must be provided when using List of Tapes or List of Instances. Optionally, DIVA can provide the Instances List in a file. Import File to provide the List of Tapes or List of Objects to migrate. The Migration Tool will not try to validate these lists before starting the migration task.

The following rules apply to the file:

- File names can have any name and extension.

  For example, `Objectname|Collection|instanceid`.

Migration Source

The Migration Source defines the source to use for the migration job. Select the check box for the source, or sources to be used for the migration job. For example, use this setting to migrate TAPES.

Migration Target

This Migration Target parameter defines the target media where objects are migrated. The valid Migration Targets are Cloud, Tape Groups and Disk Arrays. Single, or multiple targets, and the same Migration Target as the Migration Source Type can be selected. For example, use this setting to migrate TAPES.

The following rules apply to the number of new instances being created on the Migration Target:

- The Migration Tool checks how many instances of each object being migrated already exist on the target, and counts already existing instances as already migrated.

  For example, one instance for Object-A already exists on the target media, named *default*. The number of instances on the target media (*default*) is set to 1. The migration job performs no additional copy of Object-A to the target media (default) because one instance already existed before the migration task started.

- If the task has multiple media targets, and one of the targets is a Disk Array, the Migration Tool always uses instances migrated to the disk target as the Migration Source for migration to any Tape Group targets. If the alternative instance's migration fails, the original Migration Source instance is used for migration.

Migration Settings

You can use these migration settings:

- *Max Parallel Requests*—The number of concurrent requests for this migration task. Enter the number of concurrent requests to be used, or use the up and down arrows at the right side of the text box to change the number.
- *Priority Value*—This parameter identifies the priority of jobs sent to the Manager by the Migration Service during the migration task. This is a mandatory parameter.

Enter a number in the text box or use the slider bar under the Priority Value setting to increase or decrease the job priority. Select the Default Priority check box to use the default priority (50).

Migration Tool Start and Stop

The Migration Tool only performs the steps necessary to reach the intended goal, and nothing more. Any time a migration task is started, the Migration Tool examines the DIVA catalog and calculates which copy and delete operations remain incomplete. It does not matter to the Migration Tool if the job was already partially completed during a previous run, it will always adapt to the current state without the need to store any progress in a status or log file.

For this reason, a Migration Task can be freely interrupted at any time and resumed later. There are no specific actions to complete before resuming an interrupted job.

Advanced Migration Jobs

This section describes advanced migration operations.

Tape Migrations

For tape migration, the Migration Service sends a `CopyTape` job to the Manager. In each request, the third-party ID of the requests submitted by the Migration Service contains the ID of the migration task. DIVA can migrate a maximum of 100 tapes per migration task.

Migration Tasks and DIVA Jobs

- Each migration task generates a list of actions from migration task sources.

  In the case of the `CopyTapes` task, the task sources contain a list of tape barcodes. There is one action per tape barcode.

  In the case of the `CopyObjects` task, the task sources contain a list of objects. There is one action per object.

- Each task action submits one or more jobs to DIVA.

  The `CopyTapes` task supports only a single destination medium. Each action submits one `CopyTape` job to DIVA.

  The `CopyObjects` task supports a list of destination media. Each action submits a list of `CopyToGroup` jobs to DIVA, one per destination.

- When a job in DIVA aborts, the Migration Service retries only once more before aborting the task.

  The `CopyTapes` task retries only aborted `CopyTape` Jobs.

  The `CopyObjects` task re-checks only whether the destination instance already was created or not, for all actions including those that were already completed. If the destination instance was not created, the `CopyObjects` task submits a new

`CopyToGroup` job to DIVA. This results in DIVA logging already-completed actions as duplicated events for the task.

Error Handling and Failure Scenarios

This section describes how to handle errors and failures.

Migration Error Handling

Migration tasks abort if first job and second retry both abort.

You can retry both aborted `CopyTapes` and `CopyObjects` tasks by starting them again. The same retry logic applies: when the first job aborts, then a second retry happens once before the task aborts again.

Migration Failure Scenarios

This table describes specific failure scenarios, actions taken by the Migration Tool to resolve the failure, and suggested actions you can take to resolve the issue.

Failure	Migration Tool Action	Suggested User Action
The Migration Tool lost the connection to Manager.	Tries to reconnect until connection is restored.	If Manager is offline or not running, then restart Manager. If the Migration Tool cannot connect to Manager, it can be identified because new migration Requests stay too long in the Submitted state, and do not change to the Pending state. Also check the Migration Tool Logs. At this point common problems with the connection should be investigated.
The Migration Tool cannot connect to the database.	Retry the connection three times every 10 seconds; if not successful then quit. The Migration Tool connects back to the database when it is online.	Investigate and resolve common database connectivity issues, and then restart the Migration Service.
There are no online instances for an object on the Migration Source.	The Migration Tool checks for online instances only during the Initialization phase. If an object does not have any online instances, the Migration Tool skips it and assign the correct error code to the object in the database. If the object was online, but later became offline during the Initialization phase, then some requests sent to Manager will fail. The error code is stored in the database, and the job is not retried. If the requests continuously fail up to the maximum number of failures, the job pauses. If the job has multiple Source or Destination Servers, it switches to the next server. If alternative instances migrations fail, the original instances are used instead.	Browse the results of migration for each completed migration job. The results contain a list of all objects eligible for migration, and resulting error codes (if any errors occurred). Make objects accessible online again, and then rerun the migration job.

Failure	Migration Tool Action	Suggested User Action
A migration was performed from a Tape Group that has offline tapes.	If the Alternative Sources option is turned on, the Migration Tool attempts to locate disk instances for all offline instances, and then migrates them. If Recovery Mode is turned on, the Migration Tool attempts to locate disk or tape instances for offline Source Server instances, and then migrates them. If no online alternative instances are found for some offline instances, they are skipped from migration. An Object Offline error code is assigned to the objects. If the tape transitions to offline during migration, the job pauses after the maximum sequential failure requests. If the job has multiple Sources or Destination servers, it switches to the next one, and then pauses after processing them. Migration jobs for these offline tapes are marked as Aborted. The Migration Task proceeds with migrating the other TAPES that are online.	Browse the results of migration for each completed migration job. The results contain a list of all objects or tapes eligible for migration, and error codes (if any errors occurred). Make objects or tapes accessible online again, and then rerun the migration task.
The migration task was submitted with invalid parameters.	The Migration Tool assigns the Aborted status to the task.	Confirm the task parameters, and then create a new migration task with correct parameters.
A user deletes unmigrated objects from the Source Server during a migration procedure.	The Migration Tool attempts to migrate the objects, and stores the appropriate error codes in the database.	Use the DIVA Web App to Cancel or Delete the migration job. Browse the results of migration for each completed migration job. The results contain a list of all objects eligible for migration, and error codes (if any errors occurred).
A migration task was submitted with incorrect parameters.	Not applicable	Use the DIVA Web App to cancel or delete the migration task.

Failure	Migration Tool Action	Suggested User Action
There is not enough free space on the Destination.	The Migration Tool fails the configured maximum number of failed jobs sequentially, and then places the task in the aborted state.	Resolve the issue on the destination, create a new Migration Task, and run the Migration Task. .
All Migration Destination media goes offline.	The Migration Tool fails the configured maximum number of failed requests sequentially, and then places the job in the Paused state.	Resolve the issue on the destination, create a new Migration Task, and run the Migration Task.
The source tape list, instance list, or exclude list has invalid values, but the syntax is correct. For example, a tape barcode is invalid.	The Migration Tool ignores invalid values. If all of Source Server is invalid, the job must terminate.	Confirm the values entered into these lists, make any necessary corrections, and then rerun the job.

Drop Folder Monitor (DFM)

This chapter describes the Drop Folder Monitor (DFM) component of the DIVA system.

Topics

- About Drop Folder Monitor (DFM)
- Start/Stop the Drop Folder Monitor Service
- DFM Configuration
- DFM Multiple File Archive
- Archive Request Failover

About Drop Folder Monitor (DFM)

Note: Starting in DIVA 9.4, Drop Folder Monitor replaces the legacy Drop Folder Manager. The DIVA Drop Folder Monitor (DFM) allows users and third-party applications to submit content for archiving by copying related files to a folder, an FTP server, or a CIFS share.

Drop Folder Monitor (DFM) is a DIVA service executed by an actor.

DFM doesn't require an application, or user, to send the Archive command. Each time you place a file (in Single File Mode), or set of files (in File Set Mode), into a monitored drop folder, DIVA automatically archives the file(s) and creates the related object.

DFM supports many possible configuration combinations. To make configuration and operations easy, Telestream has combined the most commonly-used configurations into four basic Drop Folder Monitor solutions. Additionally, you can create more-advanced configurations.

You can start a drop folder service for a given unmanaged server. DFM can use the file-path root of the unmanaged server, or you can specify a different root directory as the drop-folder root. You can create any number of subdirectories within a drop folder.

- Directories created under the root are *collection* or *category* directories. Content dropped under a *collection* directory is archived as object with *collection* set to the name of the collection directory.
- Directories created under a collection directory are *object* directories. Files or folders dropped into an object directory are archived as a single object. ObjectName

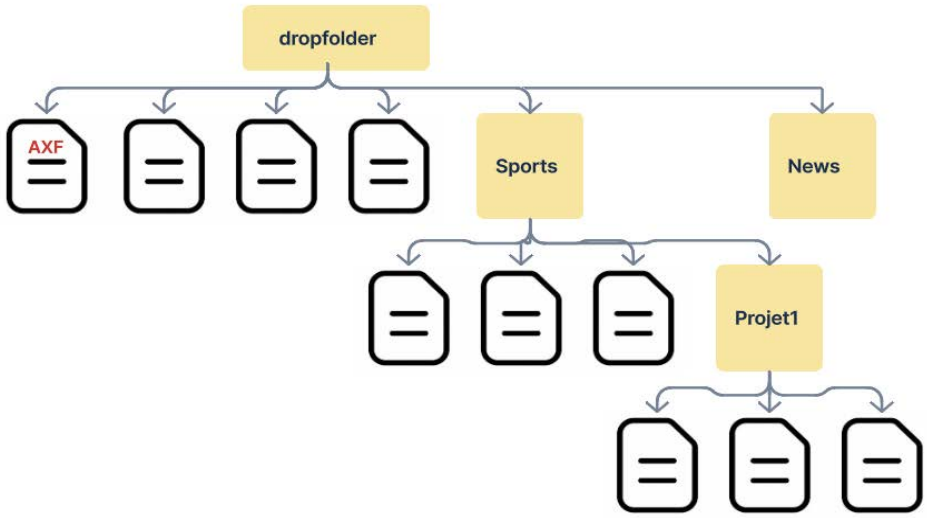
and collection are respectively set to the object-directory name and collection-directory name.

You can start a drop-folder service for a given unmanaged server. DFM can use the file path root of the unmanaged server, or a different root directory, as the drop-folder root. Different kinds of files and directory structures can be written under a drop folder. As the organization under a drop folder could be different depending on the usage, the rules to map DIVA objects from the structure are driven by a regular expression.

Supported Functionalities

- Drop Folder Monitor supports local disks and SMB network shares (SMB 3 required).
- Drop Folder Monitor can't extract the content of dropped AXF
- Drop Folder Monitor supports long path names
- Drop Folder Monitor currently runs on Windows but not on Linux

Example



DFM provides automatic monitoring of newly-created files in up to 20 local folders, FTP folders, or a combination of the two. DIVA supports one or more files in FTP folders per object. When a new file, or FTP folder, is identified, DFM issues an job for archiving the new file or folders automatically to DIVA. After the files are successfully archived, they are automatically deleted from the Source server.

You can start, stop, and restart the DFM Service using the operating-system Services or the DFM command-line utility on each host that is running a DFM installation. When the DFM Service starts, or restarts, DFM loads and validates the configuration file. If any configuration issues are detected, DIVA terminates the process and runs diagnostics.

If the configuration validation completes successfully, DFM begins scanning all of the configured drop folders, checks the status of all objects that were initialized before DFM was last shut down, and updates the internal database with the current status of the objects. After all of these checks have completed. DFM enters the Running state.

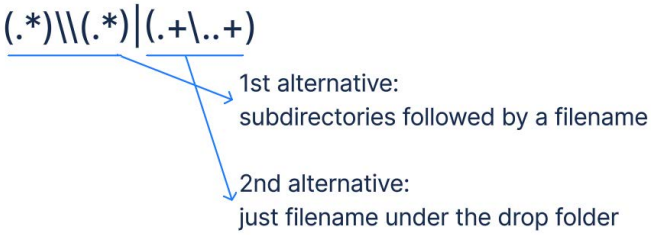
When DFM finds files in a configured drop folder, it updates the internal database and asks DIVA to archive all files found as new objects. To avoid repeated archive jobs, DFM continuously updates the archive operations status in the internal database.

If jobs fail, the Status Module informs the internal database about the failure. If the number of unsuccessful job attempts reaches a pre-configured number, the object status is changed to *could not be archived* and the object is marked as incomplete. DFM logs information about the incomplete files, and calls the DFM File DIVA Module to move them to the Trash folder.

If the job is completed successfully, the Status Module updates the internal database. In the case of a File Set Object, DFM removes the Metadata File and the File Set folder.

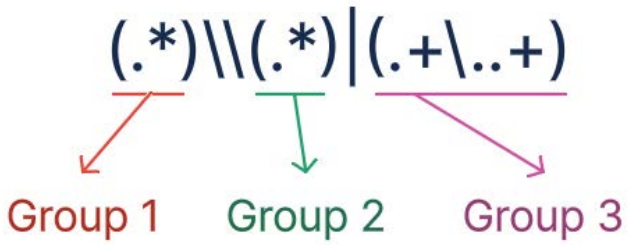
Upon execution of the shutdown script, DFM stops all internal processes before all archive operations are complete. After all of the modules are stopped, DIVA saves all internal statuses on the disk in the internal database before DFM completes the shutdown.

In the example below, if we want to archive each individual file in an object, we can use this regular expression:



Note: The second alternative expects a filename with an extension. Otherwise, top-level folders would be treated as objects.

The regular expression contains three groups between parenthesis that can be used to map the parameters of an archive request:



- The file `Sports\video1.mxf` matches the first alternative. The extracted sub-strings are:
  - Group 1: `Sports`
  - Group 2: `video1.mxf`
  - Group 3:
- The file `video2.mxf` matches the second alternative. The extracted substrings are:
  - Group 1:
  - Group 2:
  - Group 3: `video2.mxf`
- The file `Sports\Football\video1.mxf` matches the first alternative. The extracted substrings are:
  - Group 1: `Sports\Football`
  - Group 2: `video1.mxf`
  - Group 3:

In this example, we can associate objects names with the strings matched by Group 2 or Group 3. The collection can be a constant. The files path root contains Group 1. You can create more sophisticated regular expressions to rule the behavior of a given DFM service. The following fields can be constructed using the substrings matched by evaluating the relative path of each file against the regular expression: `objectname`, `collection`, `comments`, `files path root` and `object explorer path`. Telestream recommends using regular-expression test tools to anticipate the behavior of DFM.

Features

DFM monitors designated drop folders and asks DIVA to archive objects found in these folders. DFM can monitor up to 20 local folders, FTP folders, and CIFS folders; each with its own individual configuration. There are two types (modes) of drop folders; Single File Mode and File Set Mode.

For example, if the archive contains two media tape groups, News and Movies, then you must configure two drop folders; one for News and one for Movies. However, if File Set Mode is used (or an advanced configuration setup), you can use a single drop folder instead.

After archiving is complete, DFM deletes the files from the folder from which they were archived.

Caution: Copy files to the drop folder; don't move them there. After the configured time period, if the file is considered incomplete, it might get deleted. Copying the file, rather than moving it, ensures that a backup copy remains available if an unwanted deletion occurs.

When the drop folder is configured on an FTP server, all files and folders to be archived must be copied to the drop folder using FTP. If you copy the content to the drop folder using another process (for example, local copy through the operating system, remote or desktop copy) the contentmight not get processed properly.

DFM supports drop-folder configuration and monitoring from the DIVA Web App. DFM archives multi-file objects without an MDF file.

- DFM supports local disks and SMB network shares (SMB 3 is required).
- DFM can't extract the content of dropped AXF
- DFM supports long path names
- DFM works on Windows but not on Linux

DIVA Connect Web App Monitor

You can use DIVA Web App to monitor the jobs initiated by the Drop Folder Monitor. DIVA enables viewing the status of the jobs, job history, and other portions of the system. The Jobs, Job History, Catalog Browsing, and SPM Actions views under the Content Management menu offer filtering functionality. The *Content Management* > *Catalog Browsing* view displays the information in the table columns. The table columns being viewed are editable using the *Edit Table* button on the top right of the view. Use the DIVA Web App *Job History* view to query completed DFM archive jobs.

You can filter searches using the filtering icon on the top of the view. To make querying DFM archive jobs easier, assign a dedicated server exclusively for Drop Folder Monitor (possibly with the same parameters as another server in the system). However, note this technique will not work for DIVA delete jobs issued by DFM.

DFM Monitor with Logs

The Drop Folder Monitor log files must be used for status monitoring of DFM. The files are located in the actor logs under %DIVA_HOME%\Program\log\actor\<actor name>\scanningservices.log.. The log files contain detailed information about files and folders encountered by DFM, information about jobs submitted to DIVA, files/folders marked as incomplete, and errors encountered in processing.

Start/Stop the Drop Folder Monitor Service

An actor executes the Start/Stop the Drop Folder Monitor Service. For this service to be able to communicate with Manager REST API and send archive requests, do the following:

1. Browse to the *GENERAL SETTINGS* > *REST API GATEWAY* view.
2. Enter an URL into the API Gateway URL field.
3. Start DFM Services from the DIVA Web App.
   Resource Management > Unmanaged Servers.

See *Configuring an Avid Interplay Server*.

DFM Configuration

The following table provides descriptions of DFM parameters.

Parameter	Description	Default Value
Trial mode	When trial mode is set, DFM will detect files to be archived but will not submit actual archive requests.	Yes
Drop folder root	Absolute root directory to monitor on the unmanaged server. If empty, DIVA will use the files path root configured for the unmanaged server.	Empty
Inactivity Delay Before Archive	When multiple files are dropped into a subfolder of a collection directory (in an object folder), all the files is contains are archived into a single object. This delay defines how long DFM should wait without detecting activity before considering that the object is ready for archive.	
Time scheduler	When enabled, this scheduler defines timeframes when DFM must be enabled.	

Parameter	Description	Default Value
Regular Expression filter	Regular expression used to extract matching substrings from the relative path of each file added, modified, deleted, or renamed under the drop folder. Substring extraction is based on capturing groups identified by a number. The substring extracted by each group can be used to map differentiator path, objectname, collection, comments, explorer path and files path root.	
Differentiator Path mapping	Relative path that can be used to differentiate each object uniquely. When dropping a folder containing one or multiple files that must be archived as a single object, the differentiator path must be the relative path of the top level directory.	
Objectname mapping	Objectname of the objects to archive. The objectname/ collection pair must be unique. Cannot be an empty string.	
Collection mapping	Collection of the objects to archive. It can be a hardcoded string. It may include substrings extracted using the regular expression. This parameter cannot be empty. If the collection computed using matching groups, the resultant collection cannot be empty.	
Comments mapping	Comments of the objects to archive. It can be a hardcoded string, substrings extracted using the regular expression or a combination of both.	

Parameter	Description	Default Value
Object Explorer Path mapping	Virtual directory under which objects archived by DFM will be in object explorer. It can be a hardcoded string or a substring extracted using the regular expression. It can by empty if the objects are not tied to specific object explorer directories.	
Files Path Root mapping	Mapping string that is used to build the Files path root of the archive request. This path is relative to the drop folder root.	
Object directory in files path root	This parameter applies to directories (containing one or multiple files) placed in the drop folder root or in a collection directory. If this option is set to true, the object directory is in the files path root.	true
Media	Target media for the archive requests issued by DFM. Can be set to the name of a tape group, disk array or cloud bucket.	

Parameter	Description	Default Value
Priority	Priority of the archive requests issued by DFM.	50
Quality of service (QOS)	Quality of service of the archive requests issued by DFM. This parameter can be empty and the QOS will be set automatically by Manager.	0
Delete on source	When *delete on source* is enabled, DIVA will delete the archived files from the unmanaged server after a completed archive job. The selected unmanage storage must be configured to allow delete on source (-allow_delete_on_source) if this option is set to true.	false

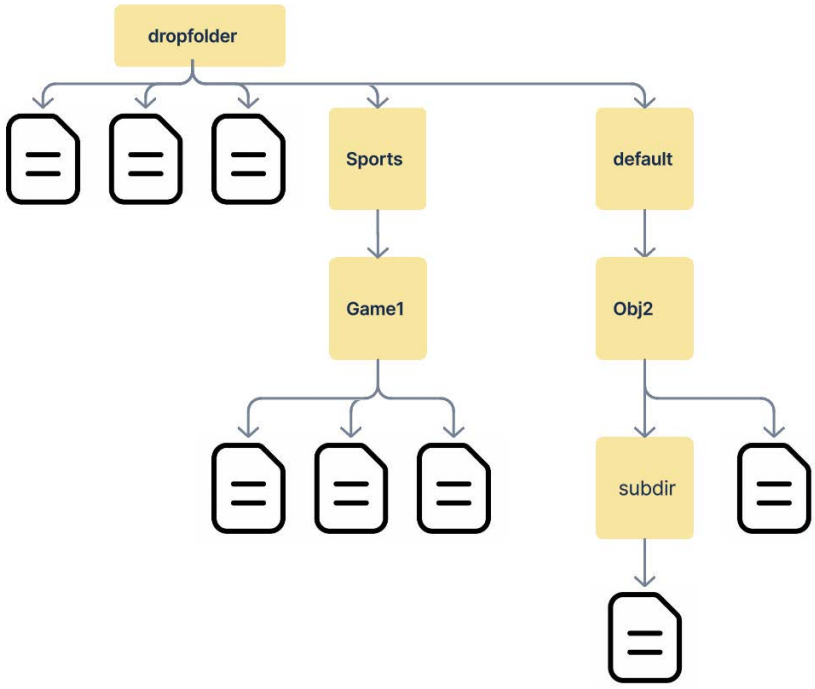
DFM Multiple File Archive

Any folder created under a collection folder is an object folder. After an observed period of inactivity, if none of the files are locked, DFM considers the content of an object folder ready to archive.

DFM considers all the files and subfolders recursively as part of the object. In the archive requests generated by DFM, the collection is set to the name of the collection folder and object name is set to the name of the object folder.

An object folder may contain a single file.

In the example below, there are two object folders that would be archived. Sports/Game1 contains three files and Default/Ojb2 contains 2 files and a subdirectory.



When DFM is set up to use *delete on source*, the object folder is deleted at the end of the archive request.

Archive Request Failover

DFM submits an archive job per file or group of files detected in the drop folder. An archive job may fail depending on the system environment or resource availability. An aborted request is retried up to five times with a growing interval between attempts. The initial interval is set to two minutes and is multiplied by the number of retries.

Replacing an Existing Object with a Dropped Object

DFM won't submit an archive job when an object already exists. Instead, DFM adds a warning to the DFM events.

To replace the existing object with a dropped object, do the following:

1. Move the object outside of the drop folder.
2. Delete the existing object in DIVA via API or the DIVA Web App.
3. Move the object back to the drop folder so that DFM can detect it again.

DFM Restart

When DFM service starts for the first time, the content of the drop folder is listed and archive requests are generated by DFM for the objectname/collectionname pairs that do not already exist. When DFM service is stopped, DIVA remembers the date/time the service was stopped. When the service is restarted, DFM considers only the content dropped 24 hours before the service was stopped.

After the initial content is processed by DFM, the service processes the new content dropped in real time.

Drop Folder Manager (Legacy)

This chapter describes the legacy Drop Folder Manager component of the DIVA system.

Topics

- About Drop Folder Manager (Legacy)
- Administration, Operations and Monitoring
- Solutions and Basic Configurations
- Additional Configuration Options
- Configuration Files and Examples

About Drop Folder Manager (Legacy)

Note: Starting in DIVA 9.4, Drop Folder Monitor replaces the legacy Drop Folder Manager. The DIVA Drop Folder Monitor (DFM) allows users and third-party applications to submit content for archiving by copying related files to a folder, an FTP server, or a CIFS share.

Drop Folder Manager allows users and third party applications to deliver content to be archived by copying related files to a folder, an FTP server, or a CIFS share.

Drop Folder Manager doesn't require an application, or user, to send the Archive command. Each time you place a file (Single File Mode), or set of files (File Set Mode), in a monitored drop folder, DIVA automatically archives the files and creates the related object.

Drop Folder Manager supports many possible configuration combinations. To make configuration and operations easy, Telestream has combined the most commonly-used configurations into four basic Drop Folder Manager solutions. Additionally, you can create more advanced configurations.

Drop Folder Manager supports multiple automated operations. Example workflow configurations follow.

- `dfm.conf.delete.ini`

  This configuration file contains the minimum configuration parameters to configure for use as a file deletion solution.

- `dfm.conf.single.local.ini`

  This configuration file is a template for configuring automated local, single file based archive operations.

- `dfm.conf.ini`

  This configuration file is a template demonstrating how multiple workflows and drop folders can coexist in a single configuration file. It is modular by design.

Features

Drop Folder Manager is a DIVA component that can you can install from the DIVA installer.

Drop Folder Manager monitors designated drop folders and asks DIVA to archive objects found in these folders. Drop Folder Manager can monitor up to 20 local folders, FTP folders, and CIFS folders; each with its own individual configuration. There are two types (modes) of drop folders; Single File Mode and File Set Mode.

For example, if the archive contains two media tape groups, News and Movies, then you must configure two drop folders; one for News and one for Movies. However, if File Set Mode is used (or an advanced configuration setup), you can use a single drop folder instead.

After archiving is complete, DFM deletes the files from the folder from which they were archived.

Copy files to the drop folder; don't move them there. After the configured time period, if the file is considered incomplete, it might be deleted. Copying the file, rather than moving it, ensures that a backup copy remains available if an unwanted deletion occurs.

When the drop folder is configured on an FTP server, all content (files and folders to be archived) must be copied to the drop folder using FTP. If the content is copied to the drop folder using another process (local copy through the operating system, remote or desktop copy, there is no guarantee it will process the content properly.

Single File Mode Drop Folders

Use Single File Mode drop folders to automatically archive objects containing a single file. This mode is convenient for MXF, or GXF, with single file assets. There can be many single files in the folder, but each will only be archived as a single file object.

The object name is the file name including the extension. The object collection, priority, and other archive command parameters are defined in a drop folder configuration file (dfm.conf). The configuration file contains the description of all folders to be monitored by Drop Folder Manager.

When a file appears in a configured drop folder, Drop Folder Manager monitors the file size. If the size has not changed after a pre-configured time threshold (last modification time), then the file is considered complete, and archiving is initiated. After the file is archived, it is automatically deleted from the drop folder.

File Set Drop Folders

Caution: The last file created in the File Set drop folder must be the metadata file with the MDF extension. The presence of this file triggers the archive operation. If the metadata file is transferred before the actual data files, content loss can occur.

File Set drop folders are used to automatically archive objects containing more than one file, and are compatible with complex objects.

When monitoring a File Set drop folder, Drop Folder Manager checks for the presence of an MDF metadata file. This file describes the details of the content to be archived, and signals DFM that the content to be archived is ready. The archive process starts only after this file is available to DFM.

Using the metadata file allows specifying the object name, object collection and user Comments on a per object basis.

Users (or an application) must first create a folder within the drop folder for the set of files intended to be archived as a single object in DIVA. The folder must have the `-allow_delete_on_source` and `-r` options configured. After the folder is created, the data files have to be copied to the folder.

The object archiving process is initiated after Drop Folder Manager has opened and parsed the metadata file. After the object is successfully archived, Drop Folder Manager deletes the folder, and all of the files within it, including the metadata file.

Common Drop Folder Manager Solutions

Telestream offers the four most common Drop Folder Manager solutions as described in the following sections. Select the solution, or solutions, that meet the requirements of the system environment, and follow the configuration instructions for those particular solutions.

Basic Solution for Single Mode Folders

This solution can monitor local disks, FTP transfers, and CIFS folders. DIVA archives each file found in the drop folder as a single object. If only single files are required then this solution will work perfectly.

- Only single file objects are supported.

- Supported major configuration parameters include:
  - DIVA
  - Drop folder URL
  - Collection Name of objects being archived
  - Server for the archive job
  - Templates for file path root and file name generation
  - Media name for the archive job

If the connection to the disk, FTP or CIFS is unavailable (or disconnected) for a period (that is, when the fileReadyThreshold time has elapsed), the file will be considered complete when it is actually incomplete.

The `fileReadyThreshold` parameter is only found in the dfm.conf.ini, and not in the dfm.single.conf.ini configuration file. However, this parameter can be manually added to the dfm.single.conf.ini file if wanted, or necessary.

Basic Solution for File Set Mode Folders

This solution offers greater flexibility than the basic solution. This configuration can monitor local disks, FTP servers, and unsecured CIFS folders containing either single files or file sets. Secured CIFS can also be used if the service is run using a suitable user account.

File Sets include a metadata file containing the details of the files included in the set. Each metadata file found in the drop folder is parsed and a new object (usually with more than one component) is archived in the DIVA system.

- Single and multiple file objects are supported.
- Recursive Archive (folder and files) is supported.
- Supported major configuration parameters include:
  - DIVA
  - Drop folder URL
  - Media name for the archive job

Basic Solution for Delete Mode

This solution monitors only the local disk, and removes files that are obsolete. If a sub-folder is empty for more than the configured allowed time period, it will also be removed. This solution is used with the other Drop Folder Manager solutions, and can be included in all Drop Folder Manager configurations.

Supported major configuration parameters include:

- Folder URL
- Interval specifying when the file will be deleted from the folder.

Basic Solution for DIVA Connect

This solution monitors only the local disk and can use either Single Files or File Sets. Each metadata file found in the drop folder is parsed and a new object (usually with more than one component) is archived in the DIVA System.

This solution is used with DIVA Connect systems that are configured with inter-site transfer capabilities. Contact your Telestream Sales Specialist for detailed information on various DIVA Connect Solutions available.

- Single and multiple file objects are supported
- Recursive archive (folders and files) is supported
- Supported major configuration parameters include:
  - DIVA
  - Drop folder URL
  - Media Name Pattern for generation of the Media Name for the archive job
  - Server Name
  - Server Path

Drop Folder Manager Component

Drop Folder Manager provides automatic monitoring of newly-created files in up to 20 local or FTP folders (or combinations of the two). DIVA supports one or more files in FTP folders per object. When a new file, or FTP folder, is identified, DFM issues an archive job automatically to DIVA to archive the new file or folders. After the files are successfully archived, they are automatically deleted from the Source server.

You can start, stop, and restart the Drop Folder Manager Service using the operating system Services or the Drop Folder Manager command line utility on each host that is running a Drop Folder Manager installation. When the Drop Folder Manager Service starts, or restarts, Drop Folder Manager loads and validates the configuration file. If any configuration issues are detected, the process terminates and runs diagnostics.

If the configuration validation completes successfully, Drop Folder Manager begins scanning all of the configured drop folders, checks the status of all objects that were initialized before DFM was last shutdown, and updates the internal database with the current status of the objects. After all of these checks have completed, Drop Folder Manager is in the Running state.

When Drop Folder Manager finds files in a configured drop folder, it updates the internal database and asks DIVA to archive all files found as new objects. To avoid repeated archive jobs, Drop Folder Manager continuously updates the archive operations status in the internal database.

If jobs fail, the Status Module informs the internal database about the failure. If the number of unsuccessful job attempts reaches a pre-configured number, the object status is changed to *could not be archived* and the object is marked as incomplete. Drop

Folder Manager logs information about the incomplete files and calls the Drop Folder Manager File DIVA Module to move them to the Trash folder.

If the job completes successfully, the internal database is updated by the Status Module. In the case of a File Set Object, Drop Folder Manager removes the Metadata File and the File Set folder.

Drop Folder Manager terminates upon execution of the shutdown script and stops all internal processes before all archive operations are complete. After all of the modules are stopped, all internal statuses on the disk are saved in the internal database before the Drop Folder Manager completes shutdown.

The Drop Folder Manager configuration file is `%DIVA_HOME%\Program\conf\dfm\dfm.conf`. Service logging is performed through the log file located in the `%DIVA_HOME%\Program\log\dfm\` folder. The logging configuration is in the `%DIVA%\Program\conf\dfm\dfm.trace` file.

The `dfm.bat` file enables managing Drop Folder Manager from a command-line interface. Execute the batch file using the following command and parameters:

```
%DIVA_HOME%\Program\dfm\bin\dfm.bat [command] [options]
```

Appending the `-conf` (or `-f`) option after one of the following commands specifies a specific configuration file to load settings from. The following table describes the `dfm.bat` parameters:

Parameter	Description
`install` (`-i`)	Installs the DFM module as a system service.
`uninstall` (`-u`)	Removes the DFM module service.
`start`	Starts the DFM module.
`stop`	Stops the DFM module if it is running.
`restart`	Stops and subsequently starts the DFM module.
`status`	Determines whether the service is running and displays the status.
`version` (`-v`)	Displays the version information and then exits.
`help` (`-h`)	Displays help information and then exits.

Administration, Operations and Monitoring

This section describes Drop Folder Manager administration, operations, and monitoring, and includes the following information:

Drop Folder Manager Overview

You can start, stop, and restart the Drop Folder Manager service using the operating system services facilities, or from the Drop Folder Manager command line utility as follows:

```
dfm {start|stop|restart}
```

The following table describes the command-line parameters:

Parameter	Description
`start`	Tells the Drop Folder Manager service to start.
`stop`	Tells the Drop Folder Manager service to stop.
`restart`	Tells the Drop Folder Manager service to stop, and then start again.

Drop Folder Manager Service Start and Stop

When Drop Folder Manager service starts, or restarts, Drop Folder Manager loads and validates the configuration file. If Drop Folder Manager detects any configuration issues, the process terminates and runs a diagnostics routine.

For successful starting of Drop Folder Manager, the INI extension must be removed from the `dfm.trace.ini` file.

If the configuration validation is successful, Drop Folder Manager scans all of the configured drop folders, then checks the status of all objects initialized before Drop Folder Manager was last shutdown, and then updates the internal database with the current status of the objects.

After all of these steps have completed, DIVA displays the Drop Folder Manager status *Running*.

Drop Folder Manager Service

Caution: The last file to be created in the File Set drop folder must be the metadata file (MDF), which then triggers the archive operation. If the metadata file is transferred before the actual data files, content loss can occur.

When Drop Folder Manager locates files in a configured drop folder, it updates the internal database and requests DIVA to archive all files found as new objects. In order to avoid repeated archive jobs, Drop Folder Manager continuously updates the archive operations status in the internal database.

Drop Folder Manager logs information about the incomplete files and calls the Drop Folder Manager File DIVA module to move them to the trash folder.

If DIVA jobs fail, the DIVA Status module informs the internal database of the failure. If the number of unsuccessful job attempts reaches a pre-configured number, the object status is changed to *Could not be archived* and the object is marked as incomplete.

If the job completes successfully, the internal database is updated by the DIVA Status module. Drop Folder Manager removes the MDF file, and the File Set folder, for a File Set object.

Drop Folder Manager Service Stop

Executing the shutdown script will terminate the Drop Folder Manager service. All internal process are stopped before all archive operations are completed. After all of the modules are stopped, all internal statuses are saved to disk in the internal database before Drop Folder Manager completes shut down.

Summary of Drop Folder Manager Administration

The following is an overview of the administrative functions available in Drop Folder Manager. There is no special licensing required to use Drop Folder Manager.

Configuring the Service

The Drop Folder Manager configuration file is `%DIVA_HOME%\Program\conf\dfm\dfm.conf`.

Upgrading the Service

Use the standard DIVA upgrade procedure to upgrade Drop Folder Manager.

Installing the Service

The Drop Folder Manager module is installed with DIVA in the same installation package.

The command line syntax to install the service is `%DIVA_HOME%\Program\dfm\bin\dfm.bat install`.

Uninstalling the Service

The command line syntax to uninstall the service is `%DIVA_HOME%\Program\dfm\bin\dfm.bat uninstall`.

Starting the Service

Execute `%DIVA_HOME%\Program\dfm\bin\dfm.bat start` to start the service from the command line, or start it from the Windows Services panel.

Stopping the Service

Execute `%DIVA_HOME%\Program\dfm\bin\dfm.bat stop` to stop the service from the command line, or stop it from the Windows Services panel.

Service Logging

The log folder is `%DIVA_HOME%\Program\log\dfm\`.

The logging configuration is `%DIVA_HOME%\Program\conf\dfm\dfm.trace`.

Monitoring the Service

Use the DFM logs, located in `%DIVA_HOME%\Program\log\dfm\`, to obtain the current status.

Command-Line Interface

Commands can be initiated using the operating system command-line. The syntax and options are as follows:

```
dfm {install|uninstall|start|stop|restart|status|version|help} [-conf|-f]
```

The parameters and options are as follows:

Parameter	Option	Description
`install`		This command installs the module as a system service.
`uninstall`		This command removes the executable as a system service.
`start`		This command starts the service.
`stop`		This command stops the service if it is currently running.
`restart`		This command stops, and then subsequently starts the service.
`status`		This command displays the current status of the service.
`version`		This command displays the module release information, and then exits.

Parameter	Option	Description
`help`		This command displays the help information, and then exits.
	-conf	This option identifies the configuration file to load the Drop Folder Manager settings from.
	`-f`	This option is an alternative to using the -conf option and performs the same function

Operations and Workflows

Drop Folder Manager constantly monitors the configured drop folders. When a file, or file set, is found in one of the configured drop folders, it is entered into the internal database (the internal database is an XML file) with the file size, date, and time when the file was originally found. After waiting for a configured period, Drop Folder Manager again checks the size, time, and date of the file (or file set) to see if any changes have occurred.

If the file in a drop folder configured in Single File Mode satisfies the applicable criteria in the Drop Folder Manager folder configuration (that is, it was not modified within the previous 60 seconds), Drop Folder Manager marks the file as complete and sends an Archive job to the DIVA system. If the file has not fulfilled the criteria for a complete file after a configured allowable period, then it is marked as incomplete and set for deletion.

Caution: The last file to be created in the File Set drop folder must be the metadata file (.mdf), which then triggers the archive operation. If the metadata file is transferred before the actual data files, content loss can occur!

In File Set Folders, after the metadata file (.mdf) is located in the drop folder, Drop Folder Manager marks all of the files in the File Set as complete, and initiates the Archive job to DIVA. If the metadata file doesn't exist after a configured period, then all files are marked incomplete and set for deletion.

By default, Drop Folder Manager tries to archive any files located in the configured and monitored drop folders. If the files can't be archived according to the Drop Folder Manager Folder configuration, they are marked as incomplete files. The different reasons for files and file sets being either archived, marked incomplete, or deleted are identified in the following sections.

The time interval for how often the drop folder is scanned is specified in the `fileReloadingThreshold` parameter in the configuration file for that specific folder.

Any files or file sets that aren't complete, archived, or both, within the specified time period are flagged as an unsuccessful archive, an incomplete object, or deleted.

The Drop Folder Manager internal database is stored as an XML file and updated after these different tasks take place. Each time Drop Folder Manager checks the configured drop folders, it compares the file date, time, and size to what was previously written to the internal database.

Source server files can't be deleted from a drop folder after an archive completes unless the appropriate server is configured using the `-allow_delete_on_source option`.

Algorithm and Workflow for Single File Mode Drop Folders

For Single File Mode folders, Drop Folder Manager uses a series of checks to identify if a file is complete and ready for processing, already processed, or incomplete.

To begin the process, Drop Folder Manager checks if the file size is equal to zero, or if the file has been marked as incomplete. If the file size does equal zero, or the file was already marked incomplete, then the file is skipped.

Next, Drop Folder Manager checks the date of the last modification to the file, and the size of the file. If the date of the last modification to the file, or the file size, has changed then the file is skipped. The `lastModificationTime` value used in this check is the local DFM time when the last modification of size, or date, was captured by DFM.

Drop Folder Manager checks if the difference between the current time (local DFM time) and the `lastModificationTime` is greater than the `fileReadyThreshold` parameter setting in the configuration file, whether the file is locked for processing, and whether the disk has free space available.

- If the difference is greater than the `fileReadyThreshold` parameter setting, the file is unknown to Drop Folder Manager, not locked for processing, and disk space is available, then the file will become known to DFM, and DFM directs DIVA to archive the file.
- If the difference is greater than the `fileReadyThreshold` parameter setting, the file is already known toDrop Folder Manager, not locked for processing, and should be deleted as incomplete, then Drop Folder Manager deletes the file.
- If the difference is greater than the `fileReadyThreshold` parameter setting, Drop Folder Manager knows about the file for longer than the `incompleteThreshold` parameter setting, and it is not locked for processing, then Drop Folder Manager marks the file as incomplete.

Note: An unknown status indicates that all required file information was not added to Drop Folder Manager. Drop Folder Manager stores the date and time information for each file, but if it doesn't have all of the required information, the file will remain unknown to Drop Folder Manager. It will only become known after all file information has been added to Drop Folder Manager for processing.

Example 1

The following is an example workflow:

1. The file test.mpg was found by Drop Folder Manager, and the date and size was stored.
2. The file test.mpg was found by Drop Folder Manager again (during the next cycle of folders investigation), the date and size was changed from the previous check. Therefore, the file will not be added as known to Drop Folder Manager.
3. The file test.mpg was found by Drop Folder Manager again (during the next cycle of folders investigation), the date and size was not changed from the previous check, the difference is greater than the `fileReadyThreshold` parameter setting, the file is unknown to Drop Folder Manager, not locked for processing, and disk space is available. Therefore, the file was added as known to Drop Folder Manager.
4. The Archive job was sent to DIVA.
5. The archive was complete.

Example 2

The following is an example workflow:

1. The file test.mpg was found by Drop Folder Manager, and the date and size was stored.
2. The file test.mpg was found by Drop Folder Manager again (during the next cycle of folders investigation), the date and size was changed from last check. Therefore, the file will not be added as known to Drop Folder Manager.
3. The file test.mpg was found by Drop Folder Manager again (during the next cycle of folders investigation), the date and size was not changed from the previous check, the difference is greater than the `fileReadyThreshold` parameter setting, the file is unknown to Drop Folder Manager, not locked for processing, and disk space is available. Therefore, the file was added as known to Drop Folder Manager.
4. The Archive job was sent to DIVA.
5. The archive was not completed. Therefore, it was terminated by DIVA for some reason.
6. DIVA attempts to archive again after `<repeatRequestDIVAThreshold>` seconds, but the job was rejected again.
7. If the difference is greater than the `fileReadyThreshold` parameter setting, Drop Folder Manager knows about the file for longer than the `incompleteThreshold` parameter setting, and it is not locked for processing, then Drop Folder Manager marks the file as incomplete. Otherwise, the system continues retrying, starting with Step 4.

Algorithm and Workflow for File Set Mode Drop Folders

For File Set Mode folders, Drop Folder Manager uses a series of checks to identify if a file is complete and ready for processing, already processed, or if it is incomplete.

To begin the process, Drop Folder Manager first checks to find out if the file size is equal to zero or if the file has been marked as incomplete. If the file size does equal zero, or the file was already marked incomplete, then the file is skipped.

Drop Folder Manager next checks the date of the last modification to the file and the size of the file. If the date of the last modification to the file or the file size has changed, then the file is skipped. The `lastModificationTime` used in this check is the local Drop Folder Manager time when the last modification of size or date was captured by Drop Folder Manager.

Drop Folder Manager checks if the difference between the current time (local Drop Folder Manager time) and the `lastModificationTime` is greater than the `fileReadyThreshold` parameter setting in the configuration file, whether the file is locked for processing, and if the disk has free space available.

If the difference is greater than the `fileReadyThreshold` parameter setting, the file is unknown to Drop Folder Manager and not locked for processing, then Drop Folder Manager directs DIVA to archive the file.

If the difference is greater than the `fileReadyThreshold` parameter setting, the file is known to Drop Folder Manager, not locked for processing, the file archived successfully, and the difference between the current time (local Drop Folder Manager time) and the archive time is greater than 5 seconds, or the file should be deleted as incomplete, then Drop Folder Manager deletes the file.

If the difference is greater than the `fileReadyThreshold` parameter setting, Drop Folder Manager knows about the file for longer than the `incompleteThreshold` parameter setting, and it is not locked for processing, then Drop Folder Manager marks the file as incomplete.

Note: An *unknown* status indicates that all required file information was not added to Drop Folder Manager. Drop Folder Manager stores the date and time information for each file, but if it doesn't have all of the required information, the file will remain unknown to DFM. It will only become known after all file information has been added to Drop Folder Manager for processing.

Example 1

The following is an example workflow:

1. The file test.mdf was found by Drop Folder Manager, and the date and size was stored.
2. The file `test.mdf` was found by Drop Folder Manager again (during the next cycle of folders investigation), and the date and size was changed from the previous check. Therefore, the file will not be added as known to Drop Folder Manager.
3. The file test.mdf was found by Drop Folder Manager again (during the next cycle of folders investigation), the date and size was not changed from the previous check, the difference is greater than the `fileReadyThreshold` parameter setting, the file was unknown to DFM. Therefore, the file was added as known to Drop Folder Manager.

4. The Archive job is sent to DIVA.
5. The archive was completed.
6. The file `test.mdf` was found by Drop Folder Manager again (during the next cycle of folders investigation), the date and size was not changed from the previous check, the difference is greater than the `fileReadyThreshold` parameter setting, the file is known to DFM, not locked for processing, the file archived successfully, and the difference between the current time (local Drop Folder Manager time) and the archive time is greater than 5000. Therefore, the file was deleted by Drop Folder Manager.

Example 2

The following is an example workflow:

1. The file `test.mdf` was found by Drop Folder Manager, and the date and size was stored.
2. The file test.mdf was found by Drop Folder Manager again (during the next cycle of folders investigation), the date and size was changed from the previous check. Therefore, the file will not be added as known to Drop Folder Manager.
3. The file `test.mdf` was found by Drop Folder Manager again (during the next cycle of folders investigation), the date and size was not changed from the previous check, the difference is greater than the `fileReadyThreshold` parameter setting, the file is unknown to Drop Folder Manager. Therefore, the file was added as known to Drop Folder Manager.
4. The Archive job was sent to DIVA.
5. The archive was not completed, and terminated by DIVA for some reason.
6. DIVA attempts to archive again after `<repeatRequestDIVAThreshold>` seconds, but the job was rejected again.
7. If the difference is greater than the `fileReadyThreshold` parameter setting, DFM knows about the file for longer than the `incompleteThreshold` parameter setting, and it is not locked for processing, then DFM will mark the file as incomplete. Otherwise the system will continue to retry, starting with Step 4.

Drop Folder Manager

The following sections describe different methods for monitoring the Drop Folder Manager service.

DIVA Connect Web App Monitor

You can use DIVA Web App to monitor the jobs initiated by the Drop Folder Manager

DIVA enables viewing the status of the jobs, job history, and other portions of the system. The Jobs, Job History, Catalog Browsing, and SPM Actions views under the Content Management menu offer filtering functionality. The *Content Management* > *Catalog Browsing* view displays the information in the table

columns. The table columns being viewed are editable using the *Edit Table* button on the top right of the view.

Use the DIVA Web App Job History view to query completed Drop Folder Manager archive jobs. Searches can be filtered using the filtering icon on the top of the view. To make querying DFM archive jobs easier, assign a dedicated server exclusively for Drop Folder Manager(possibly with the same parameters as another server in the system). However, note this technique will not work for DIVA delete jobs issued by Drop Folder Manager.

DFM Monitor with Logs

The Drop Folder Manager log files must be used for status monitoring of DFM. The files are located in the `%DIVA_HOME%\Program\log\dfm\` folder. The log files contain detailed information about files and folders encountered by DFM, information about jobs submitted to DIVA, files/folders marked as incomplete, and errors encountered in processing.

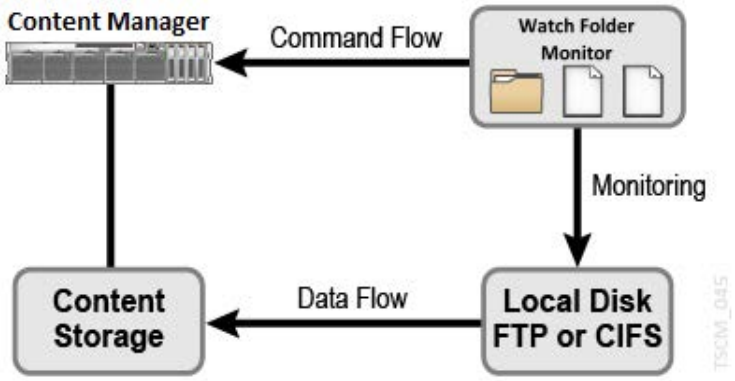
See *Trace Log Configuration* for detailed information about this file.

Solutions and Basic Configurations

This section describes details and configurations of the different Drop Folder Manager solutions, and includes the following information:

Basic Single Mode Folder Solution

This solution can monitor a local disk, FTP transfers, and CIFS folders. DIVA archives each file found in the drop folder as a single object. As files are dropped into the folder they are archived as single files into DIVA. Multiple files can be in the drop folder at the same time, but they will be handled by the Drop Folder Manager one at a time.



Supported servers include:

- Local disks

- FTP folders

  When the drop folder is configured on an FTP server, all content (files and folders to be archived) must be copied to the drop folder using FTP. If the content is copied to the drop folder using another process (local copy through the operating system, or remote desktop copy, there is no guarantee Drop Folder Manager will process the content properly.
- CIFS folders

The following are additional configuration parameters and functionality:

- Only single file objects are supported.
- Supported major configuration parameters include:
  - DIVA
  - Drop folder URL
  - Collection Name of objects being archived
  - Server for the archive job
  - Templates for file path root and file name generation
  - Media Name for the archive job

Configuring the Basic Single Mode Folder Solution

In the basic Drop Folder Manager solution, DIVA archivea each file found in the drop folder as single object. Editing the Drop Folder Manager configuration file is necessary to make the following changes.

To configure this solution for a Single Mode Folder, do the following:

1. Rename the `DFM.conf.single.local.ini` file to `DFM.conf`.

   The file is located in the `%DIVA_HOME%\Program\conf\dfm\` folder.
2. Specify the Host and Port using `<managerConnetion>` as follows (using the proper parameters):

```
<managerConnetion>
 <address host=”localhost” port=”9000”/>
</managerConnetion>
```

3. Specify the Windows Service Name using `<serviceName>` as follows:

```
<serviceName>diva_dfm</serviceName>
```

   This parameter must be written on one line, and there must not be any symbols or empty spaces at the beginning of the line.
4. Specify the Folder URL using `<url>` as follows (using the proper URL):

```
<url>file:///c:\WATCHFOLDER\</url>
```

5. Specify the Collection Name of objects to be archived using `<categoryName>` as follows (using the proper Collection Name):

```
<categoryName>Collection</categoryName>
```

6. Specify the servers to use in archive requests using `<sourceDestinationDIVAName>` as follows using the proper server name:

```
<sourceDestinationDIVAName>DISK</sourceDestinationDIVAName>
```

7. Specify the File Path to the folder containing the files DIVA archives using `<archiveFilePathTemplate>` as follows (using the proper File Path):

```
<archiveFilePathTemplate platform="DETECT" options="">
 URL_TO_FILE
</archiveFilePathTemplate>
```

8. Specify the template for the generation of the File Name using `<archiveFileNameTemplate>` as follows (using the proper Template Name):

```
<archiveFileNameTemplate platform="DETECT" options="">
 Filename.Ext
</archiveFileNameTemplate>
```

9. Specify the Media Name used in the archive request using `<divaMediaName>` as follows (using the proper Media Name):

```
<divaMediaName>Array1</divaMediaName>
```

Caution: If the following Step 10 is not completed, the installation will fail because it can't find the trace file.

10. Rename the `%DIVA_HOME%\Program\conf\dfm\dfm.trace.ini` file to dfm.trace.
11. Install the Drop Folder Manager service using the following command:

```
%DIVA_HOME%\Program\dfm\bin\dfm.bat install
```

12. Start the Drop Folder Manager service using the following command, or open the Windows Services panel and start the service from there.

```
%DIVA_HOME%\Program\dfm\bin\dfm.bat start
```

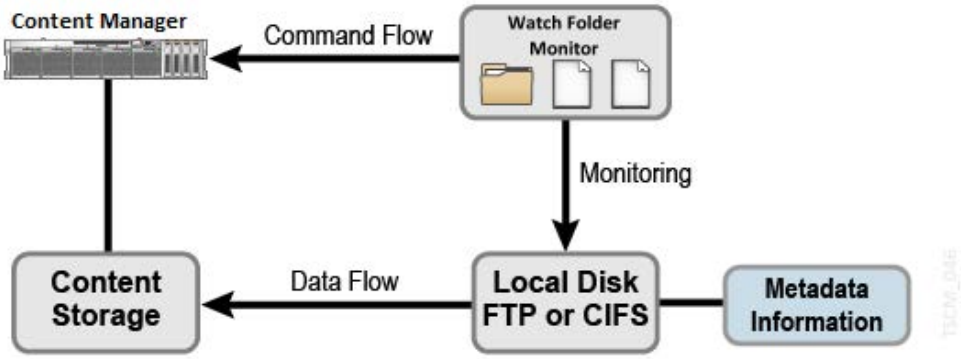
Solution for File Set Folders on Local Disk and FTP Servers

This solution offers more flexibility than the basic solution. This configuration can monitor the local disk, FTP server, and CIFS folders that contain either single files or file sets.

Drop Folder Manager supports single and set-based folder types. Each file dropped into a Single type folder results in a separate archive per file. File Set type folders provide a means of archiving objects comprised of multiple files. File Set mode archive operations require the use of metadata files, which contain the details of the file set comprising a specific archive operation. Files for separate archive operations can coexist in a File Set mode folder, because the scope of files included in each archive operation is determined by the file list in each MDF file placed in the File Set mode folder.

Caution: The last file created in the File Set drop folder must be the metadata file (with the MDF extension). The presence of this file triggers the archive operation. If the metadata file is transferred before the actual data files, content loss can occur.

AXF format files are, at the file system level, single files, but contain many files internally. AXF files can be archived into DIVA, but this requires the use of a File Set mode folder and an MDF file. The MDF file contains the list of files within the AXF file, and the target parent attributes of the object used when archived.



Supported servers include:

- Local disks
- FTP folders

  When the drop folder is configured on an FTP server, all content (files and folders to be archived) must be copied to the drop folder using FTP. If the content is copied to the drop folder using another process (local copy through the operating system, or remote desktop copy, there is no guarantee Drop Folder Manager processes the content properly.
- Secured CIFS folders

  Secured CIFS is possible if the Windows Drop Folder Manager Service is run under a suitable user account.

The following are additional configuration parameters and functionality:

- Single and multiple file objects are supported.
- Recursive archive (folders and files) is supported
- Supported major configuration parameters include:
  - DIVA
  - Drop folder URL
  - Media name for the archive job

See Metadata File.

Recursive Archive Configuration

Recursive Archive is enabled when the Drop Folder Manager folder is configured to use File Set Mode. In this mode the archive file set is defined in the specific object `configuration MDF` file.

This feature defines a special character, an asterisk, that can be used in the Drop Folder Manager metadata file (the MDF file). Applying an asterisk to the end of the folder path causes all files under this path to be archived, but doesn't include sub-folders.

Using the `-r` option in the configuration of the drop folder achieves different results than using the asterisk option. When the -r option is included in the job, the actor will recurse into all sub-folders whether the asterisk is used or not.

Conversely, if the -r option is not used, the actor only selects the files in the immediate folder. In this case, use of the asterisk will still cause archiving of all files in the immediate folder action to occur. When using this option, be aware of the job and server configurations to achieve the desired results.

Configure the -r option in either of the two following locations:

- In the job Options field when creating the job. This method is only applicable to the current job. Enter the -r option in the Options field to use this method.
- In the Server Connect Options field in DIVA. This method is applicable to all requests to this server. Include the -r option in the Connect Options field with any other request options to use this method.

Note: Recursive archive does consider the use of the `-file_order FILE_FIRST` and `-file_order DIRS_FIRST` command options.

The following are several simple examples (excerpts from the .mdf file), which enables recursive archive for the specific folders and files:

Example 1:

```
<fileList>
 folder1/*
 test/folder2/*
 folder3/test.txt
</fileList>
```

The results of this configuration are as follows:

- Every file under folder1 and test/folder2 will be archived, but not sub-folders.

  If the `-r` option is used in the job Options, or on the Server Configuration, all sub-folders and their files are also archived.
- For folder3/test.txt, only the test.txt file located in folder3 will be included.

Example 2:

```
<fileList>
```

```
 *
</fileList>
```

The results of this configuration are as follows:

- Every file in the specified folder will be archived, but not sub-folders.
- If the `-r` option is used in the job Options, or on the Server Configuration, all sub-folders and their files are also archived.

Example 3:

```
<fileList>
 folder1/*
</fileList>
```

The results of this configuration are as follows:

- Every file under folder1 will be archived, but not sub-folders.
- If the `-r` option was used in the job Options, or on the Server Configuration, all folder1 sub-folders and their files will also be archived.

Configuring the File Set Folders Solution

To configure this solution for a File Set default filter for local disks and FTP, do the following:

1. Rename the DFM.conf.set.local.ftp.ini file to DFM.conf.

   The file is located in the `%DIVA_HOME%\Program\conf\dfm\` folder.

2. Specify the Host and Port using `<managerConnetion>` as follows (using the proper parameters):

```
<managerConnetion>
 <address host=”localhost” port=”9000”/>
</managerConnetion>
```

3. Specify the Windows Service Name using `<serviceName>` as follows:

```
<serviceName>diva_dfm</serviceName>
```

   This parameter must be written on one line, and there must not be any symbols or empty spaces at the beginning of the line.

4. Specify the `<folderConfig>` node using `<type>`. The folder should be the File Set type, and therefore the set value should be specified as follows:

```
<type>set</type>
```

5. Specify the Folder URL using `<url>` as follows (using the proper URL):

```
<url>ftp://diva:diva@localhost:21/watchfolder</url>
```

6. Specify the Media Name to use in the archive request using `<divaMediaName>` as follows (using the proper Media Name):

```
<divaMediaName>Array1</divaMediaName>
```

Caution: If the following Step 7 is not completed, the installation will fail because it can't find the trace file.

7. Rename the `%DIVA_HOME%\Program\conf\dfm\dfm.trace.ini` file to dfm.trace.
8. Install the Drop Folder Manager service using the following command:

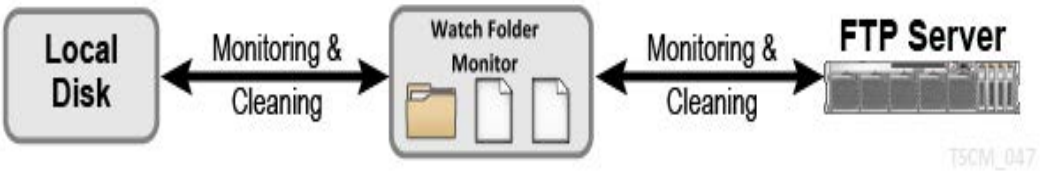
```
%DIVA_HOME%\Program\dfm\bin\dfm.bat install
```

9. Start the Drop Folder Manager service using the following command, or open the Windows Services panel and start the service from there.

```
%DIVA_HOME%\Program\dfm\bin\dfm.bat start
```

Delete Mode Solution

This solution monitors the specified folder, and removes obsolete files. If a sub-folder is empty for more than the configured allowed time period, it will also be removed. This solution is used in with the other Drop Folder Manager solutions, and should be included in all Drop Folder Manager configurations.



The following major configuration parameters are supported:

- Folder URL
- Interval specifying when files will be deleted from the folder

Configuring the Delete Mode Solution

To configure the Drop Folder Manager solution for Delete Mode, do the following:

1. Rename the `dfm.conf.delete.ini` file to `dfm.conf`.

   The file is located in the `%DIVA_HOME%\Program\conf\dfm\` folder.
2. Specify the interval that identifies when the file will be deleted from the folder using `<fileDeleteThreshold>` as follows (the default value is shown):

```
<fileDeleteThreshold>86400</fileDeleteThreshold>
```

- This value is the File Delete Threshold in seconds, and for folders only in Delete Mode.
- This interval specifies when a file will be deleted from a Drop Folder Manager folder after copying the file to the folder is complete.
- The state of completion is identified by the file size. As the file is copied, the size of the file will continue to grow until the copy is complete. Once the file size no longer increases, the system recognizes that the file copy operation is complete.
- This value must be a positive integer value.
- The default value of 86400 seconds equals 24 hours.

3. Specify the Windows Service Name using `<serviceName>` as follows:

```
<serviceName>diva_dfm</serviceName>
```

   This parameter must be written on one line, and there must not be any symbols or empty spaces at the beginning of the line.

4. Specify the `<folderConfig>` node using `<mode>` as follows:

```
<mode>Delete</mode>
```

5. Specify the Folder URL using `<url>` as follows (using the proper URL):

```
<url>file:///c:\CLEAN_FOLDER\</url>
```

Caution: If the following Step 6 is not completed, the installation will fail because it can't find the trace file.

---

6. Rename the `%DIVA_HOME%\Program\conf\dfm\dfm.trace.ini` file to dfm.trace.
7. Install the Drop Folder Manager service using the following command:

```
%DIVA_HOME%\Program\dfm\bin\dfm.bat install
```

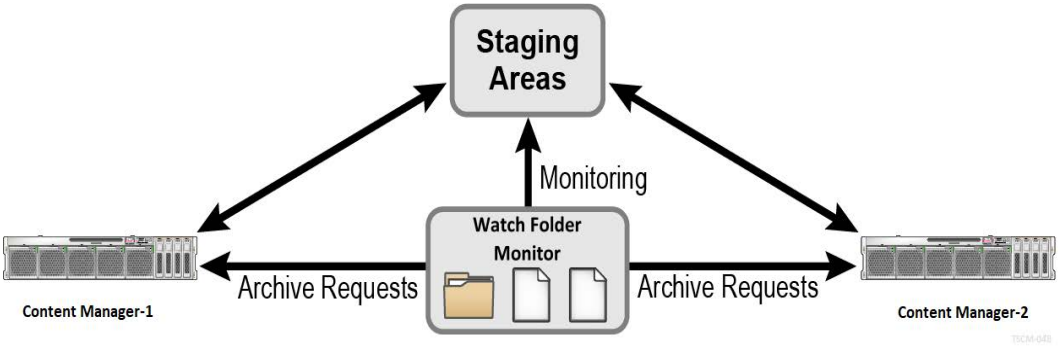
8. Start the Drop Folder Manager service using the following command, or open the Windows Services panel and start the service from there.

```
%DIVA_HOME%\Program\dfm\bin\dfm.bat start
```

DIVA Connect Solution

This solution monitors a local disk, and can use either Single Files or File Sets. Each metadata file found in the drop folder is parsed, and a new object (usually with more than one file) is archived in DIVA. DIVA Connect systems configured with inter-site transfer capabilities use this solution.

Contact Telestream Support for detailed information about various available DIVA Connect solutions.



Supported servers include:

- Local disks

The following are additional configuration parameters and functionality:

- Single and multiple file objects are supported.
  - Files must be in non-complex objects.
- Supported major configuration parameters include:
  - DIVA
  - Drop folder URL
  - Media Name Pattern for generation of the Media Name for the archive job
  - Server name
  - Server path

Configuring the DIVA Connect Solution

This solution monitors the local disk. Each metadata file found in the drop folder is parsed, and a new object (usually with more than one file but not a complex object) is archived in DIVA.

To configure the Drop Folder Manager Solution for DIVA Connect, do the following:

1. Rename the *dfm.conf.DIVA Connect.ini* file to *dfm.conf*.

   The file is located in the `%DIVA_HOME%\Program\conf\dfm\` folder.

2. Specify the Host and Port using `<managerConnetion>` as follows (using the proper parameters):

```
<managerConnetion>
 <address host=”localhost” port=”9000”/>
</managerConnetion>
```

3. Specify the Windows Service Name using `<serviceName>` as follows:

```
<serviceName>diva_dfm</serviceName>
```

This parameter must be written on one line, and there must not be any symbols or empty spaces at the beginning of the line.

4. Specify the `<folderConfig>` node using `<type>`. The folder should be the File Set type, and therefore the set value should be specified as follows:

```
<type>set</type>
```

There are two types of folders: Single and File Set. If the folder type should be Single, then the single value must be specified. If the folder type should be File Set, then the set value must be specified as shown in the example.

5. Specify the Folder URL using `<url>` as follows (using the proper URL):

```
<url>file:///c:\WATCHFOLDER\</url>
```

6. Specify the Media Name Pattern to use in the archive request using `<divaMediaNamePattern>` as follows (using the proper Media Name Pattern):

```
<divaMediaNamePattern>/data/$GROUP/</divaMediaNamePattern>
```

7. Specify the Server Name using `<sourceDestinationDIVAName>` as follows (using the proper server name):

```
<sourceDestinationDIVAName>SourceDestinationServer</
sourceDestinationDIVAName>
```

8. Specify the Server Path using `<sourceDestinationDIVAPath>` as follows (using the proper server path):

```
<sourceDestinationDIVAPath>SourceDestinationServer</
sourceDestinationDIVAPath>
```

Caution: If the following Step 9 is not completed, the installation will fail because it can't find the trace file.

9. Rename the `%DIVA_HOME%\Program\conf\dfm\dfm.trace.ini` file to dfm.trace.
10. Install the Drop Folder Managerservice using the following command:

```
%DIVA_HOME%\Program\dfm\bin\dfm.bat install
```

11. Start the Drop Folder Manager service using the following command, or open the Windows Services panel and start the service from there.

```
%DIVA_HOME%\Program\dfm\bin\dfm.bat start
```

Contact Telestream Support for more information and details about the full solution and configuration.

Multiple DFM Instances on the Same Computer

Running multiple Drop Folder Manager instances on the same computer requires a separate installation of DIVA for each additional Drop Folder Manager instance. It is possible to configure only the Drop Folder Manager component in these additional instances. The `<serviceName>` parameter must be unique among all of the Drop

Folder Manager instances. There is no special licensing required to use Drop Folder Manager in this configuration.

The following commands are for administration operations when running multiple Drop Folder Manager instances on the same computer:

Service Configuration

The Drop Folder Manager configuration file is Program/conf/dfm.conf in each DIVA folder.

Service Update

Use the standard DIVA upgrade procedure to upgrade Drop Folder Manager.

Service Installation

The Drop Folder Manager module is installed with DIVA in the same installation package.

The command line syntax to install the service using a specific configuration file is `%DIVA_HOME%\Program\dfm\bin\dfm.bat install -conf Program/conf/dfm/dfm.conf`.

Service Removal

The command line syntax to uninstall the service using a specific configuration file is `%DIVA_HOME%\Program\dfm\bin\dfm.bat uninstall -conf Program/conf/dfm/dfm.conf`.

Service Start

Execute `%DIVA_HOME%\Program\dfm\bin\dfm.bat start -conf Program/conf/dfm/dfm.conf` to start the service using a specific configuration file from the command line, or start it from the Windows Services panel.

Service Stop

Execute `%DIVA_HOME%\Program\dfm\bin\dfm.bat stop -conf Program/conf/dfm/dfm.conf` to stop the service running a specific configuration file from the command line, or stop it from the Windows Services panel.

Service Logging

The log folder is `%DIVA_HOME%\Program\log\dfm\`.

The logging configuration is `%DIVA_HOME%\Program\conf\dfm\dfm.trace`.

Service Monitoring

Use the Drop Folder Manager logs located in `%DIVA_HOME%\Program\log\dfm\`, to obtain the current status.

Additional Configuration Options

This section describes additional Drop Folder Manager configuration options and includes the following information:

Additional DFM Configuration

The following sections describe additional options for Drop Folder Manager configuration.

Incomplete File Threshold: `<incompleteThreshold>`

If the file is copied into a File Set Mode folder, and during the designated `incompleteThreshold` seconds it was not archived, the file is marked as incomplete. The default value is 86400 seconds (24 hours).

```
<incompleteThreshold>86400</incompleteThreshold>
```

Incomplete File Processing: <incompleteProcessingStrategy>

This parameter identifies how DIVA processes incomplete files. Incomplete files are files that were placed into the drop folder without the metadata file (for File Set folders), and also files that can't be archived after a specific number of attempts.

DIVA uses the following three parameters to identify incomplete files:

- `incompleteThreshold`
- `maxRejectCountInitializing`
- `maxRejectCountProcessing`

The following three options are available to use for the `OPTION` value in the parameter as shown:

- `None`

  This option directs the system to do nothing with an incomplete file.

- `Delete`

  This option directs the system to delete the incomplete file.

- `Rename`

  This option directs the system to rename the file so it is identifiable as an incomplete file.

```
<incompleteProcessingStrategy>OPTION</
incompleteProcessingStrategy>
```

MDF Configuration Priority: <mdfConfigPriority>

Note: This parameter only applies to File Set Mode folders.

This parameter can have Primary or Secondary values, and determines how the system handles the file during the archive process. This can either be by using the metadata file parameters (Primary), or the folder specific configuration parameters (Secondary). Primary is the default value used when this parameter is empty or missing.

```
<mdfConfigPriority>Primary</mdfConfigPriority>
```

Additional Operations Drop Folder Manager Configuration

The following sections describe additional Drop Folder Manager configuration options related to DIVA operations.

Delete Before Archive: <deleteBeforeArchive>

Caution: This optional parameter may lead to content deletion in the DIVA system, and must be used with caution.

This optional parameter specifies whether Drop Folder Manager should delete the object before the archive job for a specific folder. This parameter can be either true or false. The default value is false.

```
<deleteBeforeArchive>FALSE</deleteBeforeArchive>
```

Example:

An object is copied to the drop folder and fits into an established Storage Plan. The Storage Plan specifies that the object should be copied to another server, and archived to a specific folder. Using this option, the object will be copied to the specified server, but then deleted before the Archive action is performed. Therefore, the object is never actually archived.

Number of Files to Archive Wildcard: <numberFilesToArchiveWildcard>

This parameter defines a threshold on the number of files in the same folder. If the number of files in the folder is larger than the numberFilesArchiveWildcard parameter, Drop Folder Manager sends the archive job to DIVA using the asterisk wildcard as the File List parameter, instead of the actual list of files for the entire folder. The value must be a positive integer to specify the number of files, or if the asterisk wildcard is used in the job, all files will be archived.

```
<numberFilesToArchiveWildcard>*</numberFilesToArchiveWildcard>
```

Object Name Conversion Rules: <objectNameConvertRules>

Note: This parameter only applies to Single File Mode folders.

The object name conversion rules identify how DIVA translates file names into object names. Only the simple conversion method is implemented. The rule syntax for this method is case sensitive.

When specifying a rule using `Name.Ext`, the object name will be the same as the file name. If Name is only specified, the file name will be the same as the original file name, but the extension will be removed. This applies to all of the following parameters:

- `Name.Ext`

  This value results in the original file name and extension; no conversion of the file name or extension takes place.

- `name` (all lowercase)

  This value results in the original file name being converted to all lowercase, and the file extension is removed.

- `NAME` (all upper case)

  This value results in the original file name being converted to all upper case, and the file extension is removed.

- `ext` (all lowercase)

  This value results in the file name being removed, the original file name being converted to the extension, and is all lowercase.

  For example, file_name.ext will become only .ext.

- `EXT` (all upper case)

  This value results in the file name being removed, the original file name being converted to the extension, and is all upper case.

  For example, file_name.ext will become only .EXT.

- `name.Ext`

  This value results in the original file name being converted to all lowercase, and the original file extension is left intact.

- `name.EXT`

  This value results in the original file name being converted to all lowercase, and the original file extension is converted to all upper case.

- `pre_name_suf.EXT`

  This value results in the original file name being converted to pre_(original_-file_name)_suf, and the original extension is converted to all upper case.

- `pre_Name_suf`

  This value results in the original file name being converted to pre_(original_-file_name)_suf, and the extension is removed.

  For example, the following statement uses the Simple method, and leaves the file name and extension unchanged:

```
<objectNameConvertRule method="simple">Name.Ext</
objectNameConvertRule>
```

Original Server: <originalServer>

In some configurations, the monitored folder is an intermediate location between the video server and the DIVA system. The `<originalServer>` parameter can be specified to ensure the original content location is set correctly in the DIVA system. The original server is displayed in the user interface, and stored in the database as the original content location, instead of the actual server used to archive data from the intermediate location.

In the statement `<originalServer>original_servername</originalServer>`, the value of original_server is the original content location designated by the original host name. For example, the statement <originalServer>VIDEO_SERVER_1</originalServer> identifies VIDEO_SERVER_1 as the original content location.

Original Path: <originalPath>

In some configurations the monitored folder is an intermediate location between the video server and the DIVA system. Specify the `<originalPath>` parameter to ensure the content path at the original location is set correctly in the DIVA system. The original content path is displayed in the user interface, and stored in the database as the path to the original content location, instead of the actual Root Path used to archive data from the intermediate location.

In the statement `<originalPath>original_path</originalPath>`, the value of original_path identifies the content path at the original_server location.

See the previous section describing original_server.

Media Name: <divaMediaName> and <divaMediaNamePattern>

The `<divaMediaNamePattern>` is the Media Name Pattern used to select the medium for archive jobs. Using this pattern allows the GROUP value to be identified.

In the statement `<divaMediaNamePattern>folder1/$GROUP/folder3</divaMediaNamePattern>`, folder1 and folder3 are actual folder names, while $GROUP indicates the system will identify the tape group from the .mdf file.

If the pattern was not recognized, then the `<divaMediaName>` value from the file set's corresponding MDF file is used as the Media Name for the archived job when working with a File Set Mode folder. If the pattern is recognized, then the `<divaMediaName>` from the configuration file is used.

Additional Monitored Folder Configuration

The following sections describe additional configuration options related to the Drop Folder Manager monitored folder.

MDF File Extension: <mdfExtension>

This option identifies the extension for the metadata file for each folder. In the following example, the metadata file extension will be mdf:

```
<mdfExtension>mdf</mdfExtension>
```

Server Path: <sourceDestinationDIVAPath>

Note: This parameter only applies to File Set drop folders. Because the actor and Drop Folder Manager could be on different machines, they might view the folder differently. For example, actor may see the folder as a local disk, while Drop Folder Manager may see it as an FTP folder (or vice versa).

This field defines the location of the drop folder Root on the server. For a dedicated server (the Root Path of the server in DIVA points to the root of Drop Folder Manager), the File Path Root of the server may already contain a path, and then leave this field empty.

In the statement `<sourceDestinationDIVAPath>path</sourceDestinationDIVAPath>`, the value of path is the full path to the root server folder.

For example, the statement `<sourceDestinationDIVAPath>WATCH_FOLDER</sourceDestinationDIVAPath>` identifies WATCH_FOLDER as the full path to the root server folder.

Sample Server Paths for Different Configurations

In File Set Mode, the `<SourceDestinationDivaPath>` is added to the relative file path. The following examples are for different common configurations.

For more details and examples, see *File Root Path and Single File Mode Advanced Configuration*.

Example 1:

- `SourceDestinationDivaPath = E:\DFM`
- `Archive File Path Root = E:\DFM\data\tape1\123456`
- `Archive File List = toto.mdf toto0.data folder1\toto1.data folder1\folder2\toto2.data`

- Results
  - E:\DFM\data\tape1\123456\toto.mdf
  - E:\DFM\data\tape1\123456\toto0.data
  - E:\DFM\data\tape1\123456\folder1\toto1.data
  - E:\DFM\data\tape1\123456\folder1\folder2\toto2.data

Example 2:

- SourceDestinationDivaPath = F:\DFM\:
- Archive File Path Root = F:\DFM\data\tape1\123456
- Archive File List = toto.mdf toto0.data folder1\toto1.data folder1\folder2\toto2.data
- Results
  - F:\DFM\data\tape1\123456\toto.mdf
  - F:\DFM\data\tape1\123456\toto0.data
  - F:\DFM\data\tape1\123456\folder1\toto1.data
  - F:\DFM\data\tape1\123456\folder1\folder2\toto2.data

Example 3:

- SourceDestinationDivaPath = empty (no value entered)
- Archive File Path Root = data\tape1\123456
- Archive File List = toto.mdf toto0.data folder1\toto1.data folder1\folder2\toto2.data
- Results
  - data\tape1\123456\toto.mdf
  - data\tape1\123456\toto0.data
  - data\tape1\123456\folder1\toto1.data
  - data\tape1\123456\folder1\folder2\toto2.data

Example 4:

- SourceDestinationDivaPath = data
- Archive File Path Root = data\data\tape1\123456
- Archive File List = toto.mdf toto0.data folder1\toto1.data folder1\folder2\toto2.data
- Results
  - data\data\tape1\123456\toto.mdf
  - data\data\tape1\123456\toto0.data
  - data\data\tape1\123456\folder1\toto1.data
  - data\data\tape1\123456\folder1\folder2\toto2.data

Archive File Path Template: <archiveFilePathTemplate>

Note: This parameter is only applicable to Single File Mode drop folders.

This parameter specifies the rule for how the Root Path of the archive job will be generated. The platform parameter value can be one of the following:

- WIN
- SOL
- CIFS
- DETECT

The format of the value is a combination of folder names, separators, and the following keywords:

- `PARENTn`

  This keyword identifies the folder name where the definition is located.

  For example:

```
DFM_FILE_FULL_PATH = DFM_URL\PARENTn
...\PARENT_3\PARENT_2\PARENT_1\file_name.ext
```

- `URL_TO_PARENTm`

  This keyword identifies the relative path from DFM_URL to PARENTm.

- `URL_TO_FILE`

  This keyword identifies the relative path from DFM_URL to the file.

  The following is an example using this parameter:

```
<archiveFilePathTemplate platform="DETECT"
options="">URL_TO_FILE</archiveFilePathTemplate>
```

F or more details, see *File Root Path and Single File Mode Advanced Configuration*.

Archive File Name Template: <archiveFileNameTemplate>

Note: This parameter is only applicable to Single File Mode drop folders.

This parameter specifies the rule for identifying how the file names of an Archive job are generated. The platform parameter value can be one of the following:

- WIN
- SOL
- CIFS
- DETECT

The format of the value is a combination of folder names, separators, and the following keywords:

- `PARENTn`

  This keyword identifies the folder name where the definition is located. For example:

```
DFM_FILE_FULL_PATH = DFM_URL\PARENTn
...\PARENT_3\PARENT_2\PARENT_1\filename.ext
```

- `URL_TO_FILE`

  This keyword identifies the relative path from DFM_URL to the file.

- `filename.ext`

  This keyword identifies the original file name.

- `filename`

  This keyword identifies the file name without the extension.

  In the following example filename and ext represent the keywords of the pattern definition, not the sample file name and extension; filename.ext is not an actual file name.

```
<archiveFileNameTemplate platform="DETECT"
options="">filename.ext</archiveFileNameTemplate>
```

For more details and examples, see *File Root Path and Single File Mode Advanced Configuration*.

File Filter: <fileFilter>

Note: The <fileFilter> parameter is optional for the drop folder configuration. If it is not specified it in the folder configuration, Drop Folder Manager archives any file in the folder without using any masks. Only one <fileFilter> can be specified per folder.

Use this parameter to filter files to archive. The type attribute can be specified as either include or exclude.

In inclusion mode, no files are archived if there are no filters defined. Each filter enables archiving of files if they match specific rules. The file will only be archived if the file corresponds to one or more masks. If the file doesn't correspond to any masks then the file will be ignored by Drop Folder Manager, and not deleted or archived.

In exclusion mode, all files are archived if there are no filters defined. Each filter will restrict the archiving of files if they match specific rules. The file will not be archived if the file corresponds to one or more masks. If the file corresponds to any mask, the file will be ignored by Drop Folder Manager, and not deleted or archived.

The filter can be defined as a pattern where the mask has a question mark, indicating any symbol, or an asterisk indicating any string, as a Regular Expression (regexp), or both.

The following parameters are used within the <fileFilter> configuration parameter:

- `<mask>`

  This parameter specifies the pattern for the filter.

- `<regexp>`

  This parameter specifies the regular expression for the filter.

  For details on Regular Expressions visit the following websites:

  https://en.wikipedia.org/wiki/Regular_expression

  http://www.regular-expressions.info/

Delete Parent and Content Directories:

`<deleteParentDirectoryAndContentDirectories>`

This optional parameter specifies whether Drop Folder Manager should delete the parent and child folders of an MDF file. This is useful when a user places an object into a folder, and Drop Folder Manager is not set to delete the files and folders after all actions are executed on the object. Using this parameter, the object's files and folders aren't left in the folder where the object was placed. This parameter can be either TRUE (this is the default) or FALSE.

In the following example, the folders will not be deleted:

```
<deleteParentDirectoryAndContentDirectories>FALSE</
deleteParentDirectoryAndContentDirectories>
```

In the following example, the folders will be deleted:

```
<deleteParentDirectoryAndContentDirectories>TRUE</
deleteParentDirectoryAndContentDirectories>
```

Folder Configuration Examples

The examples in the following topics have had all comments that are in the actual configuration files removed to make them easier to understand.

Example of a Local Folder in File Set Mode

The object name conversion rule doesn't apply to set mode folders. Files are acted on using the original file name and extension.

In the following configuration example:

- The local folder is `C:\WATCHFOLDER`.
- The type of Drop Folder Manager folder is File Set.
- The Priority is 50.
- The Collection is Collection1.
- The Incomplete Threshold is 60 seconds.
- The Server name is LOCAL1.

- The Server path is `C:/WATCHFOLDER`.
- The Media Name is Array1.
- The Incomplete File Processing Strategy is Delete.

  All incomplete folders will be deleted.

```
<folderConfig>
 <url>file:///c:\WATCHFOLDER</url>
 <type>set</type>
 <priority>50</priority>
 <CategoryName>Collection1</CategoryName>
 <incompleteThreshold>60</incompleteThreshold>
 <sourceDestinationDIVAName>LOCAL1</sourceDestinationDIVAName>
 <sourceDestinationDIVAPath>C:/WATCHFOLDER</
sourceDestinationDIVAPath>
 <divaMediaName>Array1</divaMediaName>
 <incompleteProcessingStrategy>Delete</
incompleteProcessingStrategy>
</folderConfig>
```

Using this configuration, if a set of files is copied into the DFM monitored local folder named `C:\WATCHFOLDER`, and it has 3 files in the set with the names file1.ext, file2.ext, and `filesetdata.mdf`, they will be archived into a single file in DIVA with the following parameters:

- The Server is LOCAL1.
- The object name is `filesetdata.ext`.
- The Collection Name is Collection1.

If the metadata file is not present, the set will be considered incomplete and the files will all be deleted after 60 seconds.

Example of an FTP Folder in Single File Mode

When the Drop Folder Manager drop folder is configured on an FTP server, all content expected to be archived must be copied to the drop folder using FTP. If the content is copied to the drop folder by other means (local copy through the operating system, or remote desktop copy, there is no guarantee of proper processing by Drop Folder Manager.

In this example:

- The FTP folder is ftp://diva:diva@localhost:21/watchfolder.
- The Type of Drop Folder Manager folder is Single.
- The Priority is 20.

- The Collection is Collection2.
- The Incomplete Threshold is 120 seconds.
- The Server Name is FTP1.
- The Server Path is `/video1/data/`.
- The Media Name is Array1.
- The default object name Conversion rule for drop folders will remove the file extension.
- The `divaMediaNamePattern` is `/def/$GROUP/content`.

Using this configuration, a single file is transferred to the FTP server called *localhost*, on port 21, with diva/diva credentials, in the Drop Folder Manager-monitored FTP folder named `watchfolder`, with the file name `file1.ext`. It is archived to a single object with the following parameters:

- Server: `FTP1`
- Object Name: `file1`
- Collection Name: `Collection2`

If the file is not completely present or if the archive process doesn't complete, the file is considered incomplete and deleted after 120 seconds.

Example of a CIFS Folder in File Set Mode

This example illustrates using a CIFS folder for the drop folder. The object name conversion rule doesn't apply to set mode folders. Files are acted on using the original file name and extension.

In the following example:

- The CIFS folder is `\\tscmserver\watchfolder`.
- The Type of Drop Folder Manager folder is File Set.
- The Priority is 50.
- The Collection is Collection3.
- The Incomplete Threshold is 60 seconds.
- The Server Name is CIFS1.
- The Server Path is empty.
- The Media Name is array1.
- The incomplete file processing strategy is Rename.

All incomplete folders will be renamed to failed_ + (original_file_name).

```
<folderConfig>
 <url>file:///\\tscmserver\watchfolder</url>
 <type>set</type>
```

```
 <priority>50</priority>
 <CategoryName>Collection3</CategoryName>
 <incompleteThreshold>60</incompleteThreshold>
 <sourceDestinationDIVAName>CIFS1</sourceDestinationDIVAName>
 <sourceDestinationDIVAPath></sourceDestinationDIVAPath>
 <divaMediaName>array1</divaMediaName>
 <incompleteProcessingStrategy>Rename</
incompleteProcessingStrategy>
</folderConfig>
```

Using this configuration, a set of files is copied into the Drop Folder Manager-monitored CIFS folder named `watchfolder`, on the computer named divaserver, and has three files in the set with the names `file1.ext`, `file2.ext`, and `filesetdata.mdf`. They are archived to a single object with the following parameters:

- Server: `CIFS1`
- Object Name: `filesetdata.ext`
- Collection Name: `Collection3.`

If the metadata file is not present, the set is considered incomplete and the files are deleted after 60 seconds.

Service Log Configuration

Use the following parameters to specify the wrapper log file tracing configuration in `Program/conf/dfm/dfm.conf`.

- `wrapper.logfile.loglevel`

  This parameter specifies the level of messages to be logged.

- `wrapper.logfile.maxsize`

  This parameter specifies the maximum size of the log file in bytes, kilobytes, or megabytes.

- `wrapper.logfile.maxfiles`

  This parameter specifies the maximum log files count.

The valid logging levels are as follows:

- `NONE`

  This logging level produces no output.

- `FATAL`

  This logging level produces only messages for irrecoverable errors.

- `ERROR`

  This logging level produces all error messages.

- `WARN`

  This logging level produces all warning messages.

- `STATUS`

  This logging level produces all state changes.

- `INFO`

  This logging level produces all JVM output and informative messages.

- `DEBUG`

  This logging level produces detailed debugging information.

The following is a sample log file configuration:

```
# Log Level for log file output. (See docs for log levels)
wrapper.logfile.loglevel=INFO

# Maximum size that the log file will be allowed to grow to before
# the log is rolled. Size is specified in bytes. The default value
# of 0 disables log rolling. May abbreviate with the 'k' (kb) or
# 'm' (mb) suffix. For example: 10m = 10 megabytes.
wrapper.logfile.maxsize=10m

# Maximum number of rolled log files which will be allowed before
# old files are deleted. The default value of 0 implies no limit.
wrapper.logfile.maxfiles=4
```

Trace Log Configuration

Use the following parameters to specify the tracing configuration for Drop Folder Manager in the file Program/conf/dfm/dfm.trace:

- `LOGGING_TRACE_LEVEL`

  This parameter specifies the level of messages to be logged.

- `LOGGING_MAXFILESIZE`

  This parameter specifies the maximum size of the log file in bytes, kilobytes, or megabytes.

- `LOGGING_LIFETIME`

  The maximum lifetime of a logged item, in hours.

The following is a sample dfm.trace configuration:

```
-------------------------------------------------
# Levels can be: DEBUG, INFO, WARN, ERROR, FATAL
```

```
LOGGING_TRACE_LEVEL=INFO
LOGGING_MAXFILESIZE=7MB
LOGGING_LIFETIME=50

-------------------------------------------------
```

Advance DFM Configuration

Caution: The parameters in the following sections should not be altered unless required, and by somebody confident of having the ability to do so correctly. Misconfiguration of these parameters could result in an unstable, or non-functioning system, or both. Telestream recommends saving the original configuration file to a different file name before making any modifications to the configuration (for example, `DFM.conf.original.ini`).

Full DFM Configuration File

There are several groups of parameters for the configuration file as follows:

- Global DFM Parameters

  These parameters specify common behavior of Drop Folder Manager.
- Folders Default Parameters

  These parameters specify default values for folder-specific parameters.
- Folder Specific Parameters

  These parameters specify folder specific parameters.
- Tracing Specific Parameters

  These parameters specify tracing parameters.

To view a sample of the full configuration file, see *Full Drop Folder Manager Configuration File (dfm.conf.ini)*.

Global DFM Configuration

The Drop Folder Manager configuration XML provides the following elements for connections configuration:

- `<managerConnection>`

  This parameter specifies how Drop Folder Manager connects to DIVA. These options aren't required to be defined unless the DIVA system being connected to requires connection parameters outside of the default configuration.

  For example:

```
<address host="..." port="..." userName="..." password="..."
applicationName="..."/>
```

- `host`

  This parameter is required and identifies the address of DIVA.

- `port`

  This parameter is required and identifies the port number where DIVA is listening. This must be a positive integer value.

- `userName`

  This parameter is optional and identifies the user name to log in to the DIVA system. If the value is NULL or empty, DIVA uses the default user name and privileges from the DIVA system for this application.

- `password`

  This parameter is optional and identifies the password for the user. The connection is rejected if the provided password is incorrect.

- `applicationName`

  This parameter is optional and identifies an arbitrary string that is interpreted by DIVA as an application name. This is used for information purposes only, and can be NULL or an empty string.

- `<serviceName>`

  This parameter identifies the Drop Folder Manager Service Name for the currently installed instance. Spaces aren't allowed within the Service Name. The Service Name will be DIVA Drop Folder Manager - serviceName. If serviceName is left empty, the Service Name DIVA Drop Folder Manager is used.

- `<foibStateFile>`

  This parameter identifies the file name of the File/Object Information Base dump.

- `<doibStateFile>`

  This parameter identifies the file name of the Delete File/Object Information Base dump.

- `<fileReloadingThreshold>`

  This parameter identifies the file reloading threshold in seconds. This interval specifies how often the File Monitor module reloads the state of the Drop Folder Manager folders. This setting must be a positive integer value between 1 and 10,000,000.

- `<fileReadyThreshold>`

  This parameter identifies the time interval, in seconds. If the last modification time to a file remains unchanged after this interval, the file is marked as copied, and then archived. This setting must be a positive integer value between 1 and 10,000,000.

- `<dirDeleteThreshold>`

  This parameter identifies the Directory Delete Threshold in seconds. This interval specifies how often the File Monitor module reloads the state of the Drop Folder

Manager folders in Delete Mode, and commences with deletion. This setting must be a positive integer value between 1 and 10,000,000.

- `<fileDeleteThreshold>`

  This parameter identifies the File Delete Threshold in seconds. This interval specifies when a file will be deleted after copying is complete for Drop Folder Manager folders in Delete Mode. This setting must be a positive integer value between 1 and 10,000,000.

- `<maxArchiveOperations>`

  This parameter identifies the maximum number of simultaneous archive operations that Drop Folder Manager can perform. If the number of jobs exceeds this threshold, pending jobs must wait until all active archive operations are complete. This setting must be a positive integer value between 1 and 10,000,000.

- `<maxRejectCountInitializing>`

  This parameter identifies the maximum number of archive retries that Drop Folder Manager can perform for jobs rejected by DIVA during job initialization. Objects exceeding this limit are marked as incomplete. This setting must be a positive integer value between 1 and 10,000,000.

- `<maxRejectCountProcessing>`

  This parameter identifies the maximum number of archive retries that Drop Folder Manager can perform for jobs rejected by DIVA during job processing. Objects exceeding this limit are marked as incomplete. This setting must be a positive integer value between 1 and 10,000,000.

- `<pollingIntervalOfRequestState>`

  This parameter specifies the frequency of object status polling in seconds. This setting must be a positive integer value between 1 and 10,000,000.

- `<repeatRequestDIVAThreshold>`

  This parameter identifies the interval in seconds, before the DIVA Archive module resubmits rejected jobs. This setting must be a positive integer value between 1 and 10,000,000.

Folders Default Configuration

The Folders Default configuration parameter values are used for folders in a folder specific configuration when a parameter is not specified. All default values can be overridden within the folder specific configuration. The Folders Default configuration accepts the following parameters:

- `<type>`

  There are two types of folders; Single File and File Set. To configure the folder for Single Files, specify single. To configure the folder for File Sets, specify set.

- `<mode>`

  This parameter identifies the Drop Folder Manager folder work mode. This can be Archive or Delete.

- `<priority>`

  This parameter identifies the default priority level of the jobs for the folder. The folder priority range is 0 - 100. The value 0 is the lowest priority, and 100 is the highest.

- `<CategoryName>`

  This parameter identifies the default value for the Collection Name. This value is provided to DIVA if it is not overridden in the folder specific configuration, or in the File Set metadata file.

- `<incompleteThreshold>`

  This parameter identifies the Incomplete Threshold in seconds. It specifies the maximum amount of time that a Single File, or File Set, can reside in the drop folder and not be archived before it is marked as incomplete. This must be a positive integer value between 1 and 10,000,000.

- `<sourceDestinationDIVAName>`

  This parameter identifies the Server Name to use in the Archive job.

- `<sourceDestinationDIVAPath>`

  This parameter identifies the Server Path to use in the Archive job.

- `<mdfExtension>`

  This parameter identifies the extension of the metadata file for each folder.

- `<originalServer>`

  This parameter identifies the server to store in the object catalog if different from the actual server. This is useful when using an intermediate storage area for archiving, but storing the original source for the data in the object catalog information.

- `<originalPath>`

  This parameter identifies the File Root Path to store in the object catalog, if different from the actual File Root Path. This is useful when using an intermediate storage area for archiving, but storing the original source for the data in the object catalog information.

- `<incompleteProcessingStrategy>`

  This parameter identifies the incomplete folder and file processing strategy. The value can be None, Delete, or Rename.

- `<numberFilesToArchiveWildcard>`

  This parameter identifies the threshold for the number of files in the same folder. If the number of files in the folder becomes larger than the `<numberFilesToArchiveWildcard>` value, Drop Folder Manager sends the Archive job to DIVA using the asterisk wildcard for the entire folder. This value must be a positive integer.

- `<divaMediaNamePattern>`

  The Media Name Pattern specifies the archive medium by extracting a part of the archive Root Path. This option is mutually exclusive with the *`<divaMediaName>`* option.

- `<deleteBeforeArchive>`

  This parameter specifies whether DFM should delete the object from DIVA before the archive operation, if the object already exists. This parameter can be set to either TRUE or FALSE. This is an optional parameter and the default value is FALSE.

- `<objectNameConvertRule method="simple">Name.Ext</objectNameConvertRule>`

  The value of `<objectNameConvertRule>` specifies the rule for the algorithm specified. In DFM, only the Simple conversion method is implemented.

  For the rule syntax for this method, see the topic, Object Name Conversion Rules: `<objectNameConvertRules>`.

Folder Specific Configuration

All folders monitored by Drop Folder Manager must be described in the configuration file. Folder specific configuration accepts the following parameters:

- `<url>`

  This parameter identifies the Drop Folder Manager folder URL. This is the monitored folder as viewed by the Drop Folder Manager process.

- `<type>`

  There are two types of folders; Single File and File Set. To configure the folder for Single Files, specify single. To configure the folder for File Sets, specify set.

- `<mode>`

  This parameter identifies the Drop Folder Manager folder work mode. This can be Archive or Delete.

- `<priority>`

  This parameter identifies the default priority level of the jobs for the folder. The folder priority range is 0 - 100. The value 0 is the lowest priority, and 100 is the highest.

- `<CategoryName>`

  This parameter identifies the default value for the Collection Name. This value will be provided to DIVA if it is not overridden in the File Set metadata file.

- `<incompleteThreshold>`

  This parameter identifies the Incomplete Threshold in seconds. This parameter specifies the maximum amount of time that a Single File, or File Set, can reside in the drop folder and not be archived before it is marked as incomplete. This must be a positive integer value between 1 and 10,000,000.

- `<sourceDestinationDIVAName>`

  This parameter identifies the Server Name to use in the Archive job.

- `<sourceDestinationDIVAPath>`

  This parameter identifies the Server Path to use in the Archive job.

- `<divaMediaName>`

This parameter identifies the Media Name to use in the Archive job. This option is mutually exclusive with the *<divaMediaNamePattern>* option.

- `<mdfExtension>`

  This parameter identifies the extension of the metadata file for each folder.

- `<originalServer>`

  This parameter identifies the server to store in the object catalog if different from the actual server. This is useful when using an intermediate storage area for archiving, but storing the original source for the data in the object catalog information.

- `<originalPath>`

  This parameter identifies the File Root Path to store in the object catalog, if different from the actual File Root Path. This is useful when using an intermediate storage area for archiving, but storing the original source for the data in the object catalog information.

- `<incompleteProcessingStrategy>`

  This parameter identifies the incomplete folder and file processing strategy. The value can be None, Delete, or Rename.

- `<numberFilesToArchiveWildcard>`

  This parameter identifies the threshold for the number of files in the same folder. If the number of files in the folder becomes larger than the Number of Files to Archive Wildcard: <numberFilesToArchiveWildcard> value, Drop Folder Manager sends the Archive job to DIVA using the asterisk wildcard for the entire folder. This value must be a positive integer.

- `<divaMediaNamePattern>`

  The Media Name Pattern specifies the archive medium by extracting a part of the archive Root Path. This option is mutually exclusive with the *<divaMediaName>* option.

- `<deleteBeforeArchive>`

  This parameter specifies whether Drop Folder Manager should delete the object from DIVA before the archive operation, if the object already exists. This parameter can be set to either TRUE or FALSE. This is an optional parameter and the default value is FALSE.

- `<objectNameConvertRule method="simple">Name.Ext</objectNameConvertRule>`

  The value of <objectNameConvertRule> specifies the rule for the algorithm specified. In Drop Folder Manager, only the Simple conversion method is implemented.

  For the rule syntax for this method, see the topic, Object Name Conversion Rules: <objectNameConvertRules>.

File Root Path and Single File Mode Advanced Configuration

The Drop Folder Manager configuration receives file information from the remote or local media (FTP server, shared directory, or mounted drive. Based on the file name and folder, Drop Folder Manager generates the File Root Path and file name.

Long path names are supported on Windows. Absolute path names are supported on Windows to a maximum of 4000 characters. Relative path names are limited to 256 characters on Windows systems (only).

Because requirements for the file name vary, the system can either include or exclude, the parent folder in the file. The requirements for the path also vary because there are several ways the actor can connect to the media.

The File Root Path must be specified in Single File Mode in the following format:

```
<archiveFilePathTemplate platform=”DETECT”
options=””>URL_TO_FILE</archiveFilePathTemplate>
```

The file name must be specified in Single File Mode in the following format:

```
<archiveFileNameTemplate platform=”DETECT”
options=””>filename.ext</archiveFileNameTemplate>
```

Drop Folder Manager monitors a Drop Folder Manager URL (FTP, CIFS, Windows disk, or Solaris disk). Drop Folder Manager obtains a File Full Path (for FTP, CIFS, Windows disk, or Solaris disk) for each incoming file.

```
File Full Path = DFM_URL\PARENTn
...\PARENT3\PARENT2\PARENT1\Filename.Ext
```

or

```
File Full Path = URL_TO_PARENTm\PARENTm
...\PARENT3\PARENT2\PARENT1\Filename.Ext
```

Note: The operating system determines whether to use / or \ for these statements.

```
<archiveFilePathTemplate platform=”DETECT” options=””>x:\ccc\ddd</
archiveFilePathTemplate>
```

The format of the value is a combination of folder names, separators, and keywords for `PARENTn`, `URL_TO_FILE`, and `URL_TO_PARENTm`.

```
<archiveFileNameTemplate platform=”DETECT”
options=””>filename.ext</archiveFileNameTemplate>
```

The format of the value is a combination of folder names, separators, and keywords for `PARENTn`, `URL_TO_PARENTm`, `URL_TO_FILE`, `Filename.Ext`, and `Filename`, where the platform is WIN, SOL, CIFS, or DETECT.

Available keywords are as follows:

- `PARENTn`

  This keyword identifies the folder name where the definition is located. For example:

```
DFM_FILE_FULL_PATH = DFM_URL\PARENTn
...\PARENT_3\PARENT_2\PARENT_1\Filename.ext
```

- `URL_TO_PARENTm`

  This keyword identifies the path from DFM_URL to PARENTm.

- `URL_TO_FILE`

  This keyword identifies the relative path from DFM_URL to the file.

- `Filename.ext`

  This keyword identifies the original file name.

- `Filename`

  This keyword identifies the file name without the extension.

It is important to use the correct case in all of the following examples because the file name and extension parameters are case sensitive. For example, filename.ext is different than Filename.Ext.

Workflow A

The files are located in the root of DFM_URL.

- The drop folder is on CIFS
- The actor is on Windows

  DFM_URL = `file:///\\aaa\bbb\ccc\ddd`

  File Full Path = `\\aaa\bbb\ccc\ddd\fffffff.txt`

- The actor identifies the Drop Folder Manager URL as `x:\ccc\ddd`
- The configuration should be:

```
ARCHIVE_FILE_PATH_TEMPLATE = "x:\ccc\ddd"
ARCHIVE_FILE_NAME_TEMPLATE = "Filename.Ext"
```

- `Filename.Ext` must support both files without the extension, and specifies keeping the extension if there is one.

If some files are located in the root of the `DFM_URL`, and some are located in subfolders, the path length is not constant.

This can be addressed by using `ARCHIVE_FILE_NAME_TEMPLATE = "URL_TO_FILE\Filename.Ext"`. When the `URL_TO_FILE` parameter is empty, DFM automatically removes the backslash character. An empty `ARCHIVE_FILE_PATH_TEMPLATE` value is supported.

Workflow B

If Drop Folder Manager monitors the video server using FTP, the file must have a file extension, and the actor will archive through the server interface without a file extension.

- The file must be located in a single folder level

- The drop folder is on FTP

  When the Drop Folder Manager drop folder is configured on an FTP server, all content expected to be archived must be copied to the drop folder using FTP. If the content is copied to the drop folder by other means (local copy through the operating system, or remote desktop copy, there is no guarantee of proper processing by Drop Folder Manager.

- The actor is using a special interface

  DFM URL = `ftp://user:password@aaa/bbb/ccc/ddd`

  File Full Path = `/aaa/bbb/ccc/ddd/eee/fffffff.txt`

- The actor sees the Drop Folder Manager URL as `"special_#*:/ddd"`
- The configuration will be as follows:

```
ARCHIVE_FILE_PATH_TEMPLATE = "special_#*:/ddd/URL_TO_PARENT1"
ARCHIVE_FILE_NAME_TEMPLATE = "PARENT1/Filename.Ext"
```

- This is how the file extension is removed.

Configuration Files and Examples

This section includes default configuration files and configuration examples, and includes the following information:

Object Configuration

All File Set Objects are described by a companion MDF (Metadata File). Object configuration must be specified in the MDF file using the following parameters:

Parameter	Description
`priority`	This parameter identifies the priority of the DIVA job for this object. Object priority can be in the range of 0 - 100. The value 0 is the lowest priority and 100 the highest.
`CategoryName`	This parameter identifies the object collection Name. Drop Folder Manager provides this value to DIVA.
`objectName`	This parameter identifies the object name. Drop Folder Manager provides this value to DIVA.
`comments`	This parameter identifies optional information describing the object. Drop Folder Manager provides this value to DIVA.

Parameter	Description
`sourceDestinationDIVAName`	This parameter identifies the Server Name used in the archived job.
`sourceDestinationDIVAPath`	This parameter identifies the Server Path used in the archived job.
`originalServer`	This parameter identifies the server to store in the object catalog, if different from the actual server. This is useful when using an intermediate storage area for archiving, but storing the original source for the data in the object catalog information.
`originalPath`	This parameter identifies the File Root Path to store in the object catalog, if different from the actual File Root Path. This is useful when using an intermediate storage area for archiving, but storing the original source for the data in the object catalog information.
`fileList`	This parameter identifies the list of File Path Names that should be archived relative to the folder specified in `sourceDestinationDIVAPath` parameter.

Metadata File

Caution: The last file to be created in the File Set drop folder must be the metadata MDF file, which then triggers the archive operation. If the metadata file is transferred before the actual data files, content loss can occur!

Create the MDF (Metadata File) in the original MDF format, or in XML format. Initially, Drop Folder Manager attempts to parse the MDF as an XML file. If it fails, then it will parse it in the original MDF format.

Sample of the sample.mdf File in XML Format

The following is an example of the MDF file in the XML format:

```
#
#Object configuration.
#
```

```
<DIVAObjectDefinition>
 <priority>50</priority>
 <CategoryName>Collection1</CategoryName>
 <objectName>object name</objectName>
 <comments>here are our comments</comments>
 <sourceDestinationDIVAName>SD1</sourceDestinationDIVAName>
 <sourceDestinationDIVAPath>/video1/data/</
sourceDestinationDIVAPath>
 <originalServer>original_server</originalServer>
 <originalPath>original_path</originalPath>
 <divaMediaName>array1</divaMediaName>
 <fileList>
 <file>object1/1.mid</file>
 <file>object1/2.mid</file>
 <file>object1/3.mid</file>
 </fileList>
</DIVAObjectDefinition>
```

Sample of the sample.mdf File in Original MDF Format

The following is an example of the MDF file in the original format:

```
#
#Object configuration.
#
priority=50
CategoryName=Collection1
objectName=object name
<comments>
 here are comment lines
</comments>

sourceDestinationDIVAName=SD1
sourceDestinationDIVAPath=/video1/data/

originalServer=original server
originalPath=original path
```

```
<fileList>
 object1/1.mid
 object1/2.mid
 object1/3.mid
</fileList>
```

Full Drop Folder Manager Configuration File (dfm.conf.ini)

This file must be configured and renamed to *dfm.conf* to use Drop Folder Manager. You can find an example of the full Drop Folder Manager *dfm.conf.ini* configuration file here: `\Program\conf\dfm`.

Proxy Management

This chapter describes DIVA proxy management.

Topics

- About Proxy Management
- Installing Proxy Service
- Importing a Proxy
- Playback Hosting for Proxies and Images
- Workflow Initiator

About Proxy Management

The proxy service generates video proxies within the DIVA ecosystem. These proxies are stored as objects which contain the proxy, the related keyframes, and the metadata acquired during the transcoding process. The proxy service generates proxy objects in the background, and detects when objects require proxy generation.

You can configure the proxy service to generate proxies automatically. This generation operates on a configured time schedule which restricts generation to particular days, times, or both, to avoid conflicting with the normal operation of DIVA. By default, the schedule is unrestricted.

By default, the configuration file is located here:
`(Root of DIVA)\Program\conf\proxy\Proxy.settings.json`

You can modify this path during service installation.

Installing Proxy Service

To Install Proxy Service run `proxy.bat` from the command line. The default installation for DIVA will install Proxy Service on the server running the Manager. Use the following syntax:

```
proxy.bat [parameter] [options]
```

The following table describes the parameters and options.

Parameter	Option	Description
`install (or -i)`		Installs the module as a system service.
	`-log`	Path to log directory. Default: ..\..\log\proxy
	`-conf`	Path to configuration directory. Default: ..\..\conf\proxy
	`-httpport`	Port to listen for http connections. Default: 1976
	`-httpsport`	Port to listen for https connections. Default: 1977
	`-certpath`	Path to certificate located on disk.
	`-user`	Username to install the service under. Blank entries are installed as LocalSystem.
	`-pass`	Password for the provided user.
`uninstall (or -u)`		Removes the executable as a system service.
`start`		Starts the module.
`stop`		Stops the module if it is currently running.
`restart`		Stops and then starts the module.
`status`		Determines whether or not the module is running.
`version (or -v)`		Displays the module version information and exits.
`help (or -h, or -?)`		Displays this information and exits.

Importing a Proxy

You can import proxies into DIVA from any source that can generate a CSV proxy file. Generally, you export the CSV proxy file from the source; then, you can import the proxy data from the CSV into DIVA, via the DIVA Web App.

To import a proxy, do the following:

1. In the DIVA Web App, browse to *Configuration* > *Resources* > *Proxy Storage* > *Proxy Import*.

   DIVA opens the *PROXY IMPORT* view.
2. Click *Upload Task*.

   DIVA opens the *PROXY IMPORT* view.

   DIVA opens the UPLOAD IMPORT PROXY CSV dialog.
3. Drag the CSV file into the window in the dialog.
4. Click *SUBMIT*.

Creating a Proxy CSV Manually

To manually create a proxy CSV file that DIVA can import, create a CSV file in the following form:

```
name,category,path,som,csv
"<object name>","<collection/category>","<path to the mp4 to import>","<start time if not 00:00:00:00>","<Metadata CSV contents if you have the original CSV metadata from the parent object>"
```

Exporting a DIVAView Proxy CSV

You can export the proxy mappings from DIVAView, version 6.1.0.228 or later, into a CSV file. You can then import this proxy-mapping CSV file into DIVA via the DIVA Web App. To export a proxy CSV file from DIVAView, do the following:

1. Make sure the path exists for the CSV file you wish to create.
2. Do one of the following:

- Log into `PGAdmin` and execute the following command:
- Use `Psql` to execute the following command:

```
SELECT export_mp4_paths('<path to the file you would like to create'>);
```

Proxy Import

With the DIVA Web App, you can import proxy-mapping CSV files from any source into DIVA.

Preparation for Importing Proxies

Before beginning an import process, do the following:

1. Ensure that the DIVA Proxy Service has access to the source disks that contain the proxies you are attempting to import.

   The import process uses a power-shell script with administrative privileges. The script has inherited access, depending on the rights of the user that the proxy service is running as.

2. Download the `FFMpeg` binaries from `Releases · BtbN/FFmpeg-Builds`.

   `ffmpeg-n6.1.1-93-g9593b727e2-win64-lgpl-6.1.zip` has been tested, but you can use a later build if needed.

   Make sure that the lgpl version of `ffmpeg` is the one downloaded.

3. Install the binaries in a directory that the proxy service can access.

4. Browse to this script file:
   `C:\DIVA\Program\Proxy\Packages\Tools\FFMpeg\bin\PrepareProxyImport.ps1`.

5. Copy this script file to the bin directory of `FFMPeg`, and make note of the path to this file.

   You need this path for configuration.

Configuring Proxy Import

To configure proxy import, do the following:

1. Browse to *Configuration* > *Proxy Storage* > *Configuration* > *IMPORT SETTINGS*.

   The following table provides descriptions of the settings on the *CONFIGURATION* view.

Proxy Configuration Setting	Description
Staging Source	This refers to the unmanaged storage to stage imported proxies to so they can be archived.
Concurrent Import Jobs	This is the number of concurrent archives that the proxy import task will run.
Replace Existing Proxies	This will determine whether or not to overwrite a proxy that already exists for a given object in the import task.

Proxy Configuration Setting	Description
Delete Original Import File	This will determine whether or not to delete the original import file or to preserve it. Note: When importing from DIVAView, make sure that this is disabled since the export will provide paths directly to the proxies on DIVAViews proxy storage.
Use Workflow Initiator	This flag indicates whether to use the workflow initiator service to import the supplied proxies.
Workflow Engine	This allows the selection of configured workflow engines within the workflow initiator.
Workflow Name	This is the workflow used to produce the proxy.
Import Mappings	These mappings determine replacement paths and share credentials for the uploaded CSV. If you import mappings, you don't have to manually replace paths within your export to conform to a path that is accessible from your DIVA system. You can have as many import mappings as necessary.

2. (Optional) Configure import mapping on the configuration view. These mappings determine replacement paths and share credentials for the uploaded CSV.

   This eliminates having to manually replace paths within your export to conform to a path that is accessible from your DIVA system. Configure as many import mappings as necessary to complete the import.

3. The following table provides descriptions of the Import Mappings settings.

Import Mapping Setting	Description
Import Path	This is the target path within your CSV to either apply credentials or mask with the mask path parameter.
Mask Path	This is the path relative to your DIVA system that you would like to overwrite the import path with. If this is a local path, no credentials will be necessary. If this is another remote path that requires specific access, credentials will need to be filled out.
User Account	This is the account to gain access to the share you are attempting to import from. This supports either user@domain or domain\user syntax, as well as simply user which will be a local machine account to the DIVA system.
Password	Password for the user account.

4. Browse to *Configuration* > *Proxy Storage* > *Proxy Import*.

5. Click *Upload Task*.

DIVA opens the *UPLOAD IMPORT PROXY CSV* dialog.

6. Do one of the following:

- Click *BROWSE* and browse to the desired proxy CSV file.
- Drag the desired proxy CSV file, and drop it into the *Drag File Here* window.

Note: The import process accepts only a single task at a time. If there is a task already loaded delete it first before uploading a new one.

7. Click *SUBMIT*.

Once a task is uploaded, DIVA lists import jobs and their states. This list is search-able using the filters provided in the component. You may also retry and skip jobs as necessary.

8. If you are satisfied with the list of jobs, click *Start Import*.

DIVA imports the designated proxy CSV file.

9. (Optional) To pause this process, click *Stop Import*.

Configuring Workflow Service Module (WFS)

The Workflow Initiator service enables a robust integration between DIVA and Telestream Vantage, and between DIVA and a large variety of encoders.

To configure WFS, do the following:

1. In the DIVA Web App, select *Configuration* > *Services* > *Workflow*.

   DIVA opens the *WORKFLOW* > *CONFIGURATION* view.
2. Click the Configuration tab, and select the options desired.

Task Trimming. Task Trimming runs at midnight daily, deleting tasks older than the number-of-days threshold.

Edit Engine Accessible Arrays. Accessible Arrays allows the WFS to read the source clip from the archive if it is accessible, so that the Manager doesn't make a temporary copy on the Transcoder cache.

Selectable arrays must be disk arrays with format AXF_RF_1_1.

You can make this optimization with Transcode Archived and Restore Transcode job requests, but not with Archive Transcode job requests.

When this optimization is used, for example in a Transcode Archived job, DIVA logs an event in the request job events.

Edit Engine Fallback Host and Fallback Port. DIVA uses the Fallback Engine if the executing task contains AAF files, and a host name or IP address is set in the field. The fields are visible in the user interface only if the Engine type is Vantage. If it isn't possible to differentiate Avid content from regular content by collection name, with the fallback engine, WFS can still fall back to MTE for proxy generation.

If a Vantage task is submitted where the input has any AAF files, and fallback engine is set, DIVA queues it in MTE instead. It will use any other settings from Vantage for MTE. Manager will have copied the files to the cache of the main engine; so either MTE has access to this path or the job fails.

You can have either a Vantage engine and an MTE engine, and control which one is used through proxy filters; or you can have a Vantage engine with an MTE fallback configured. Then WFS decides whether to use Vantage as requested, or the fallback MTE engine.

3. Click *Save*.
4. To see the status of the Workflow Service, click the *System Health Status* button.

   See *System Health Status Button*.

See *Transcoder Name Option (*`-tr_names`).

Playback Hosting for Proxies and Images

Proxies and images are hosted from the proxy service. You can configure multiple disks under the proxy array. This allows for more flexible configurations. To add disks to a disk array, do the following:

1. Browse to *Configuration* > *Proxy Storage* > *Arrays*.
2. Select the desired array.
3. Click the *Add Disk* (+) button to the right of *DISKS IN PROXY ARRAY*.
4. Configure the disk as needed.

Workflow Initiator

The Workflow Initiator is an option for transcoding. Use the Workflow Initiator as the default means of importing. Workflow Initiator supports most importation formats, though it employs a resource-intensive process.

Note: The current Telestream Vantage workflow does a full transcode rather than remuxing.

Storage Policy Manager

This chapter provides information on the Storage Policy Manager (SPM).

Topics

- About Storage Policy Manager
- Installation and Configuration
- Operations

About Storage Policy Manager

The Storage Policy Manager (SPM) provides life-cycle management for all objects under management in DIVA, and is typically installed on the same server as DIVA.

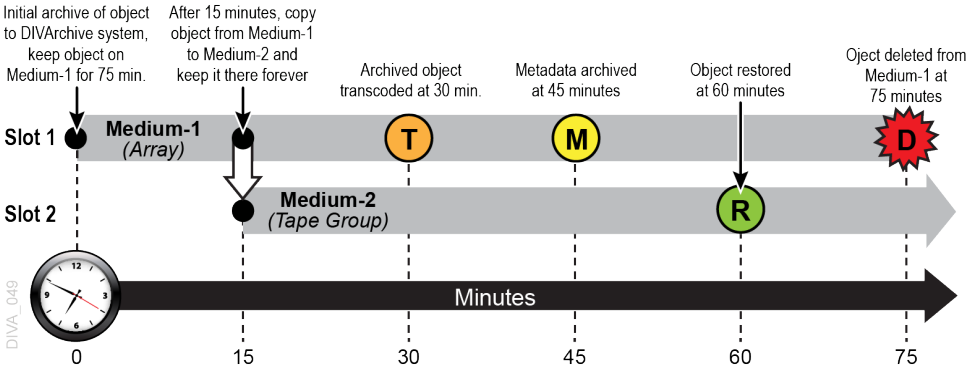
For example, an archived object can reside on a specific medium the first day, and migrate (over time) to a different medium according to established policies and rules. DIVA performs object life cycle migration as a background activity by following the rules and policies defined in the corresponding storage plan.

Caution: Misconfiguration of SPM may lead to unexpected and disastrous results. Minor edits to slots can lead to catastrophic consequences including the deletion of hundreds of thousands of instances on tape (for example, an incorrect value in a slot's end time). Without special training and familiarity with the product, always contact Telestream Support before making any changes to SPM. Failure to do so may result in severe damage to the DIVA system or permanent data loss.

The Object Life Cycle in DIVA

Each object in DIVA has its own life cycle: Standard life cycle steps and processes control each object managed by DIVA object life cycle. Understanding these steps and processes, along with terms specific to SPM helps you use DIVA more effectively.

The purpose of SPM is to automate the generic object life cycle, to reduce manual labor and potential errors. There are several steps in the object life cycle, depicted in this figure:



DIVA can define a storage policy for each object, or group of objects, which enables describing the complete life cycle of an object, and managing the content migration throughout the life cycle. Using the Storage Policy Manager, content life cycle management becomes a background process that is performed automatically.

A Storage Plan is used to specify the different migrations, and copies of the specified objects, from one media to another according to defined storage rules and policies.

The SPM component in DIVA manages the Storage Plan. The Storage Plan is a way to select the best trade off between cost and performance of the different storage media technologies as a function of the age and access frequency of the object. The Storage Plan feature enables the customization of content life cycle management.

Rules and policies can be specified that include parameters such as:

- The amount of time an object is to be retained on a medium. This interval is referred to as a *slot*.
- Where to copy an object when its time has expired on the medium (for example, copy from array to tape after two weeks).

  For example, schedule tasks that consume relatively large amounts of system resources during low use periods (for example, between midnight and five am). These tasks are executed within the interval specified in the corresponding slot configuration.
- Time of day (local time) when the action is executed.
- When and where to restore an object.
- When to transcode an object, and where to store the transcoded content.

SPM Slot Types

SPM has a logical workflow as follows:

- A Medium is a combination of defined medium (disk arrays or tape groups) available to SPM.

- Each Storage Plan is associated with one, or more, mediums and contains one, or more, slots.
- Each defined Filter is associated with a Storage Plan.
- Each slot defines one, or more, Actions to be performed.

Note: Before any configuration changes transpire, you must pause SPM. Browse to *Configuration* > *Resources* > *Storage Plans*.

To create a slot, navigate to the *Configuration* > *Resources* > *Storage Plans* > *Slots* view in theDIVA Web App and click the + button on the top of the tab.

To edit an existing slot, navigate to the *Configuration* > *Resources* > *Storage Plans* > *Slots* view in DIVA Web App, and then click the three buttons and select *Edit* from the resulting menu for the slot you want to edit in the Slots panel. Alternatively, click the name of the slot to edit.

The following is a list of Slot Types and Actions.

Storage Slot

A Storage Slot enables copying of an object, and then deleting the object (if the slot is configured for deletion). Object deletion refers to either a *DeleteInstance* or *DeleteObject* process, depending on the SPM configuration, and whether other instances still exist. Create and configure Storage Slots on the DIVA Web App *Configuration* > *Resources* > *Storage Plans* > *Slots* view.

See *Simple SPM Configuration Example* for information on storage slot workflows, and *Configuring Storage Slots* for information on configuring storage slots.

Transcode Archive Slot

A transcode archived slot enables transcoding of an archived object. Create and configure this type of slot using the DIVA Web App *Configuration* > *Resources* > *Storage Plans* > *Slots* view.

See *Action Slot Workflow* for information on Action Slot workflows, and *Configuring Transcode Archived Slots* for information on configuring transcode archived slots.

Metadata Archive Slot

A metadata archive slot enables you to archive object metadata. Create and configure this type of slot using the DIVA Web App
Configuration > *Resources* > *Storage Plans* > *Slots* view.

See *Action Slot Workflow* for information on Action Slot workflows, and *Configuring Metadata Archive Slots* for information on configuring metadata archive slots.

Restore Slot

A Restore Slot enables restoring of an object. Create and configure this type of slot using the DIVA Web App *Configuration* > *Resources* > *Storage Plans* > *Slots* view.

See *Action Slot Workflow* for information on Action Slot workflows, and *Configuring Restore Slots* for information on configuring Restore Slots.

Actions, Action Steps, and Action States

A single Storage Plan can contain a single slot, or multiple slots. Each slot has one or two actions associated with it.

Every action has one or two steps associated with it as follows:

- Copy, Transcode Archive, Metadata Archive, and Restore

  These slots have a single step associated with them.

- Delete (only valid for Storage Slots)

  The Delete Action has two steps associated with it as follows:

  – POSTPONED

    Before SPM executes any Delete Actions, it checks whether the medium is watermarked. If the medium is marked Yes for Watermarking, and the level has not reached the High Watermark, SPM will not execute the Delete Action immediately. Instead, SPM marks the object for deletion, and sets the object state to POSTPONED.

    If the medium is not watermarked, there will only be one step (Delete), and the object will be deleted immediately upon expiration.

  – Delete

    When the medium is marked Yes for Watermarking, and the watermarked medium level reaches the High Watermark, then the object will actually be deleted.

All slot details are managed from a single view, and all Slot Types only require configuration from the DIVA Web App Slots tab.

When an object is added to DIVA, SPM checks for object compliance with the filters specified in the system. When an object conforms to one of the configured filters, the filter determines which configured Storage Plan to use for processing of the object. An object can be assigned to only one Storage Plan. If the object conforms to multiple filters, the first filter the object conforms to is the one that is considered.

The slots associated with the identified Storage Plan determine the actions performed on the object. If an object did not conform to any configured filters, the object is assigned to the SP_DEFAULT Storage Plan. The SP_DEFAULT is the default Storage Plan and must have no slots associated with it.

Action States indicate the status of the action. Each action performed on an object goes through different states as shown in the following example. Each state will finish processing before the status is updated to the next state.

Example:

1. An object matches the filter for a specific Storage Plan.
2. The Storage Plan's slot schedules the associated actions for execution on the object; the status is now SCHEDULED.
3. The action is then loaded into the Action Queue; the status is now LOADED.
4. The action is now executed; the status is now PROCESSING.
5. Upon successful execution of the action the status is updated to COMPLETED.

The additional possible action status states are POSTPONED, FAILED LONG, and REJECTED. The following table describes these additional statuses. For additional Slot Workflow information, see *Storage Policy Manager Workflows*.

The following are the different SPM Action States:

Action State	Description
SCHEDULED	This state is the initial state of an action. The action is scheduled and will be loaded into the Action Queue for execution.
LOADED	The action is loaded into the Action Queue.
PROCESSING	The action is being executed. The action is loaded into the Action Queue and SPM has started processing this action for execution.
POSTPONED	When a Delete Action is encountered on a watermarked array, SPM will mark the object for deletion, and set the state to POSTPONED. SPM will not actually delete the object until the High Watermark level is reached, and then the object will be removed from the array. the object will be deleted immediately if the medium is not watermarked. This state only applies to Deleted Actions.
COMPLETED	The action has completed processing successfully.
FAILED LONG	The action failed, and will be retried according to the SPM Configuration. See the `UPDATE_ACTIONS_RETRY_FAILED_DELAY` parameter in *SPM Configuration File Parameters*.
REJECTED	SPM sets an action to the REJECTED state when it has reached the maximum number of retries and has failed. Rejected actions are never retried again. SPM also sets Copy Actions of a Storage Slot to the REJECTED state if the target medium already has instances whose quantity is greater than the instances value configured in the Storage Slot.

Watermarks and Arrays

You can manage deletions on disk mediums using watermarks. When watermarks aren't used, deletions occur immediately after the slot expires. When watermarks are

used, deletions are postponed until the disk array's occupied space hits a configured watermark.

The DSM (Disk Space Monitor) is a function of SPM, and monitors SPM identified arrays, not individual disks. The DSM process only starts if there is a SPM array that is configured for watermarking.

When an object is set for deletion by a Storage Slot, it is not actually deleted until the watermarked array reaches the High Watermark. After the array reaches the High Watermark, objects marked for deletion will be deleted either by last access time, or largest object size. the method selected depends on how the watermark is configured. Objects are then deleted until the low watermark level is reached.

Objects on watermarked arrays are deleted using one of two methods:

- Last Access Time

  This method deletes the oldest objects, among the objects marked for deletion (in non-Mixed Mode), first according to the last time the object was accessed.

- Largest Object Size

  This method deletes the largest objects, among the objects marked for deletion (in non-Mixed Mode), first according to the object size.

DSM has checks the Array watermark levels through the DIVA REST API. Requests are sent to the DIVA and the watermark levels are sent back to DSM.

SPM arrays have three possible Watermark settings that are configurable in the DIVA Web App:

- Yes

  This setting watermarks the identified array, and only considers objects already marked for deletion.

- No

  This setting doesn't watermark the identified array.

- Mixed

  This setting incorporates a combination of watermarked and not watermarked. This setting is only valid for Storage Slots. The action taken depends on which of the following events occurs first:

  – The slot reaches its end time.

  – The High Watermark is reached. This considers both objects marked for deletion, and objects whose slots are still open.

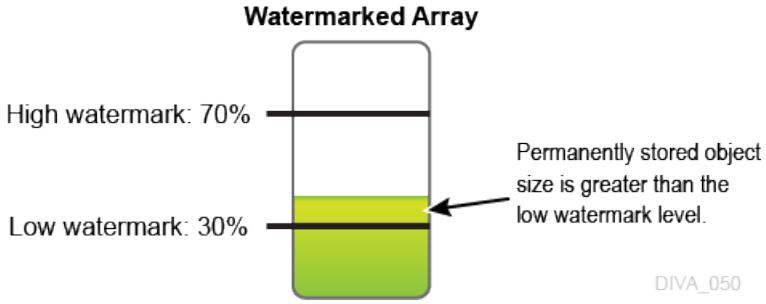
Note: SPM must be restarted if the watermark state of an array is changed.

Best Practices for Watermarking

If the total size of all permanently stored objects (that is, objects with no specified expiration time) on an array is greater than the configured Low Watermark, Telestream

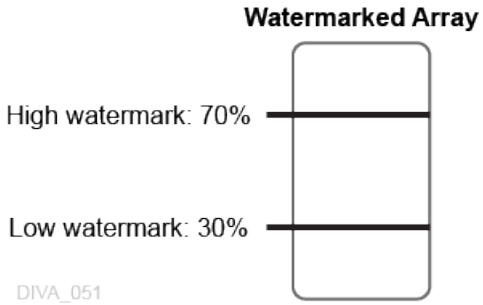
recommends reconfiguring the Low Watermark setting to a value higher than the amount of permanently stored objects.

The following figure represents this basic concept:

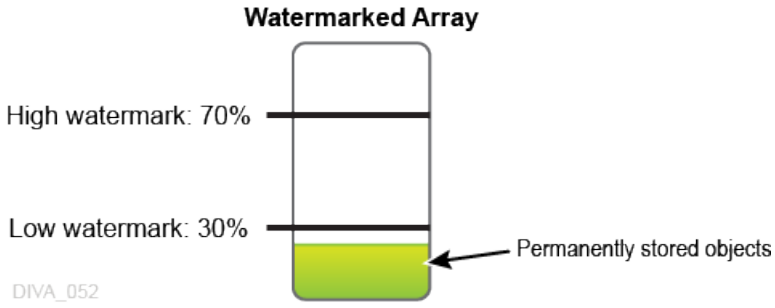


The following figures characterize a typical Watermarked Medium (watermarking set to Yes) and illustrate how watermarking ensures the medium doesn't reach full capacity.

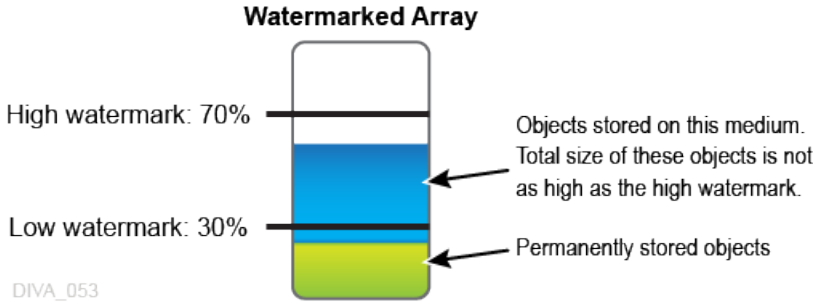
The following figure represents a watermarked disk array with no objects, and shows the Low Watermark and High Watermark:



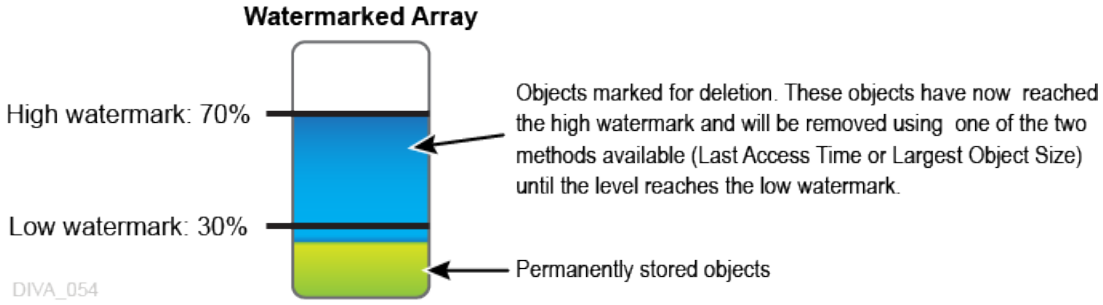
The following figure represents a watermarked disk array with permanently stored objects with a total size that is less than the Low Watermark:



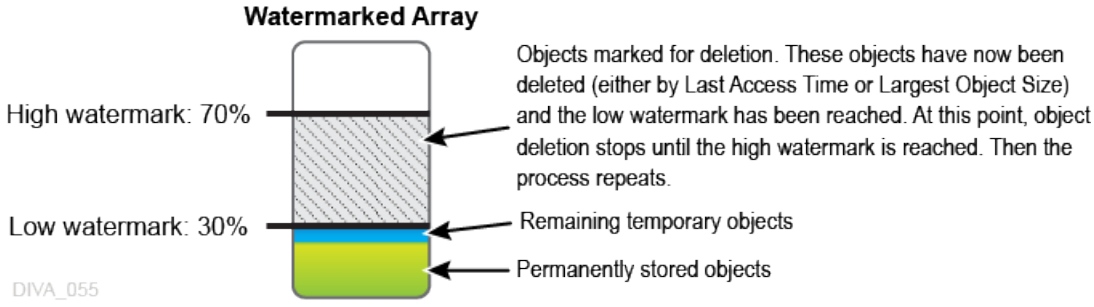
The following figure represents a watermarked disk array with permanently stored objects with a total size that is less than the low watermark, and temporarily stored objects with a total size that is higher than the low watermark:



The following figure represents a watermarked disk array with permanently stored objects with a total size that is less than the low watermark, and temporarily stored objects with a total size that has reached the high watermark level. in this figure the temporary objects have now reached the high watermark, and will be removed using one of the two methods available (last access time or largest object size) until the level reaches the Low Watermark.



The following figure represents a watermarked disk array after the temporary objects have been deleted down to the Low Watermark. Objects have now been deleted (either by last access time or largest object size) and the Low Watermark has been reached. At this point, object deletion stops until the High Watermark is reached again; then the process repeats.



Storage Policy Manager Workflows

The following sections describe the Storage Policy Manager workflows.

Storage Policy Manager Tasks

The SPM module runs several processes, each in charge of a particular task, and uses the DIVA database to process actions. There are always multiple tasks processing in parallel when SPM is operational. All actions currently being executed by SPM reside in the Action Queue until the execution is complete, and then they are deleted from the queue. Actions in the queue can be viewed using the DIVA Web App SPM Actions panel. The DIVA Web App must at least be connected to the DIVA database to access this view.

The following list describes internal tasks used by SPM:

- Update

  This task is responsible for generating, or updating, the actions based on the SPM configuration so the Load task can load the actions into the Action Queue.

- Load

  This task loads the actions into the Action Queue from the database for processing.

- Execution

  This task executes the actions loaded into the Action Queue, and submits jobs to DIVA as necessary.

- DSM (Disk Space Monitor)

  DSM monitors a particular array (not individual disks) to make sure it doesn't exceed the High Watermark value.

- Recovery

  When SPM starts, the Recovery task checks for the actions that are in an inconsistent state (LOADED or PROCESSING) and sets them to the SCHEDULED state. The action will be in an inconsistent state when SPM was terminated, or stopped during execution of the actions. If the rescheduled action has a DIVA Request ID associated with it, it will be loaded into the Action Queue as part of the Recovery task so that SPM can update the Action Status based on the status of the job submitted to DIVA.

  An action will be in an inconsistent state and have a Request ID associated with it if SPM was terminated, or stopped, when SPM:

  – Has executed the action

  – Submitted the job to DIVA

  – Was waiting for the status of the job

  The Recovery task only runs during SPM startup. If there are no actions in an inconsistent state, the task will just end.

Simple Object Life Cycle Example

The following is a very simple object life cycle for an object being processed through SPM. The process for this example life cycle is as follows:

1. An object exists on the Video server.
2. The object is archived to a disk array that is known to DIVA, and is one of the SPM monitored mediums.
3. SPM processing now begins as follows:
   a. SPM checks if the object matches any of the SPM Filters.
   b. If the object matches one of the filters, processing by SPM continues by identifying which Storage Plan is associated with the corresponding filter.
   c. When the Storage Plan is identified, SPM detects the slots that are included in the managed Storage Plan.
   d. The slots contain the actions that will be performed on the object.
   e. SPM processes the detected actions (included in the slots) on the object until processing is complete, or the slot ends (*closes*).
4. According to the Storage Slot that was created for the Storage Plan in the example, 10 minutes after the object is archived, it is copied to a target tape group.
5. SPM deletes the object from the DIVA system and SPM 10 minutes later (20 minutes after initially being archived to DIVA).

SPM Failed Action Retry and Issue Resolution

When a job action fails to execute properly, or a connection issue occurs, SPM will retry the action according to the following scenarios. The initial retry is automatic, and the job action state remains as PROCESSING.

Note: The retrying connections to the API and database are now different due to the REST API and the new SPM database pooling. However, in practice, a similar result may occur.

- DIVA Connection Failures

  SPM uses the DIVA API to connect to DIVA. If the DIVA connection is down, SPM continues retrying to establish the connection to DIVA until a connection is established.

- Database Connection Failures

  SPM will continue retrying to connect to the DIVA database every 20 seconds until a connection is established.

- Missing Instances

  The following example indicates what occurs if SPM is looking for an instance that doesn't exist:

  A Storage Slot is configured with Once Only set to N. The slot starts five minutes after archiving, and ends 30 minutes after archiving.

- SPM finishes the Copy Action after 5 minutes.
- A user manually deletes the copy made by SPM after 10 minutes.

  SPM senses the deletion and creates the copy again, because the slot period has not ended yet.
- The copy made by SPM is manually deleted again after 35 minutes.

SPM doesn't perform another Copy because the slot period has already ended.

If the Once Only parameter was set to Y, SPM will only make the copy one time. If the copy made by SPM is manually deleted, SPM will not make the copy again.

Action Retries

The following list describes when SPM will retry failed actions.

For Action Slot and Storage Slot workflows, see *Action Slot Workflow* and *Storage Slot Workflow*.

- DIVA is Busy

  If SPM could not execute the actions by submitting a job because DIVA is busy executing more jobs than the value configured in `DIVA_MANAGER_MONITOR_MAX-_REQUESTS` in the SPM.settings.json configuration file, SPM will retry the same action after few seconds delay. The delay value is configured using the `DIVA_MANAGER_MONITOR_ACTION_DELAY` parameter in the SPM.settings.json configuration file.
- Action Failed_Long State

  When SPM executes an action by submitting a job to DIVA and the job fails, the action is marked as FAILED LONG. SPM will retry the failed action after the delay period configured in the `UPDATE_ACTIONS_RETRY_FAILED_DELAY` parameter in the SPM.settings.json configuration file. SPM continues retrying this action 1,000 times using the configured delay. If the action continues to fail, it is marked as REJECTED and is never retried again.

  After placed in a Failed_Long state, only Copy, Delete, and Restore Actions are retried. All other actions will not be retried.
- Action REJECT State

  SPM sets an action to the REJECTED state when it has reached the maximum number of retries and has failed. Rejected actions are never retried again. SPM also sets Copy Actions of a Storage Slot to the REJECTED state if the target medium already has instances whose quantity is greater than the instances value configured in the Storage Slot. This is a permanent failure.
- Once Only Slots

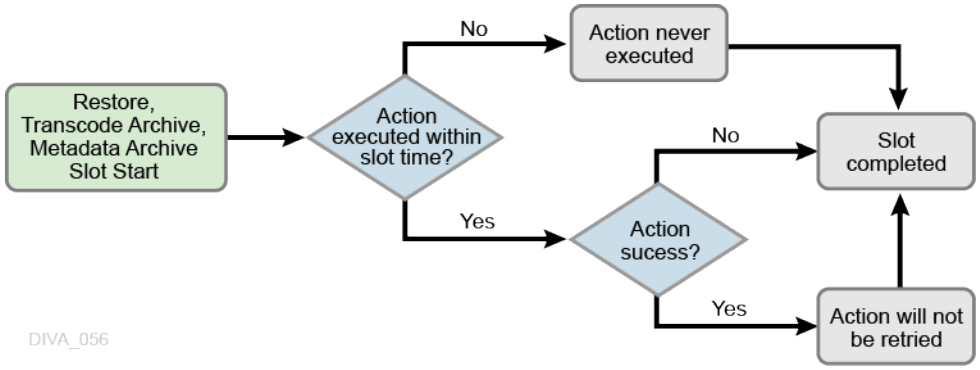
  If the Once Only parameter is set to false for a Storage Slot, all of its actions will be retried throughout the Slot Start Time and Slot End Time period.

Action Slot Workflow

Action Slots have a specific workflow that encompasses what DIVA will do in certain cases. The flowchart below displays the typical Action Slot workflow. There are three possible outcomes:

- Success
- Failed Long
- Not executed within the configured slot time

The following figure depicts the Action Slot workflow:

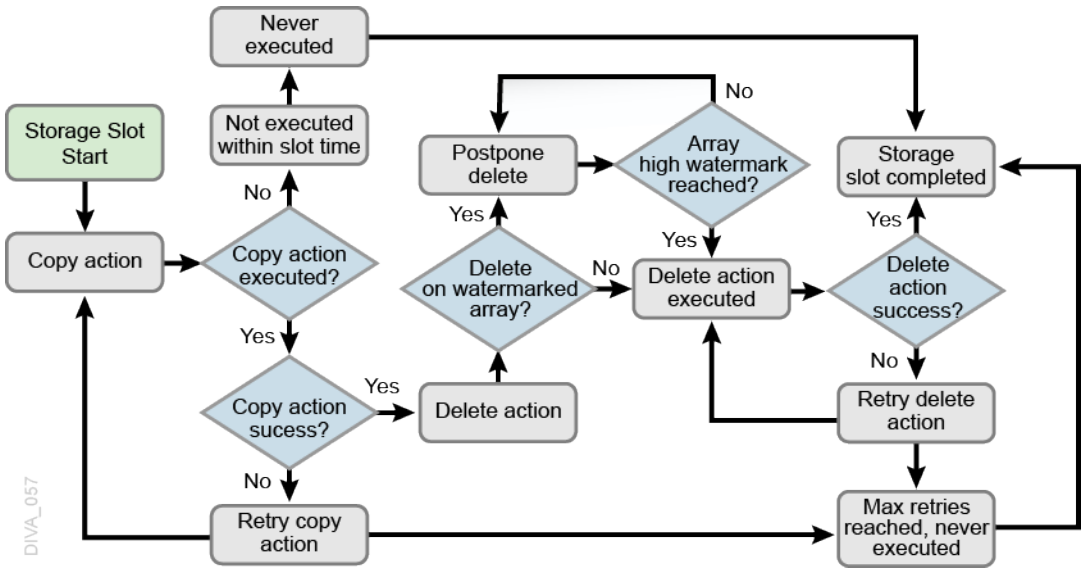


Storage Slot Workflow

Storage Slots also have a specific workflow that encompasses what DIVA will do in certain cases. The flowchart below displays the typical Storage Slot workflow. There are two possible outcomes:

- Success
- Failed Long

The following figure depicts the Storage Slot workflow:



SPM Best Practices

The following sections describe Telestream recommended best practices for configuring and using the Storage Policy Manager.

SPM Configuration

This section identifies recommended best practices for SPM configuration.

Typical SPM Workflow

A typical SPM workflow consists of archiving objects to disk, and then copying them to tape. In this type of workflow, the SPM is configured to keep the disk copy for a fixed time (non-watermarked mode), or until no more space is available (watermarked mode). Keeping a disk copy for some period allows faster access to the data.

For Action Slot and Storage Slot workflows, see *Action Slot Workflow* and *Storage Slot Workflow*.

Non-Watermarked (Fixed Retention) Mode

In Non-Watermarked Mode (Fixed Retention) Mode, the disk copy is kept until the Slot End Time (for example, two 2 days from the Archive), at which time the disk instance is deleted; the corresponding DeleteInstance SPM action is sent to DIVA, and then updates the action to the Complete state.

For more information on watermarking arrays, see *Watermarks and Arrays*.

Watermarked Mode

In Watermarked Mode, the disk instance is not deleted when the Slot End Time is reached. Instead, it is flagged as expired by setting the corresponding DeleteInstance

SPM action to the POSTPONED state. When the disk's space use reaches a configurable High Watermark (for example, 90%), SPM will remove as many expired instances from the disk as required to lower the space use down to the configured Low Watermark (for example, 60%).

In Watermarked Mode, Telestream advises not to set the disk's Slot End Time for an extended period. The reason for this is so that SPM doesn't run short of expired instances when a disk purge is triggered. However, the Slot End Time should be far enough in the future to allow the tape copy to complete. If a tape copy isn't available at the time of the DeleteInstance, the action will update to a FAILED status instead of POSTPONED. SPM will refuse to expire an instance on disk when no alternate instance is available.

A common disk Slot End Time for Watermarked Mode is three hours (180 minutes), or 24 hours (1440 minutes) if the slot's schedule only allows copies to be executed at a particular time. Confirm that the array is large enough to accommodate additional archived instances for the slot's configured time. If 100 Gigabyte of data per hour is archived at peak time, and the Slot End Time is configured to three hours, the array should be 500 Gigabyte (or larger).

For more information on watermarking arrays, see *Watermarks and Arrays*.

Choosing Appropriate Watermarks

Watermarks must be chosen carefully from the start. However, proper values are typically obtained after spending some time observing the system behavior. Telestream recommends starting with common values, and then fine-tuning them later. Typical common starting values are:

- HWM (High Watermark) = 90%
- LWM (Low Watermark) = 75%

The watermarks refer to the usage ratio of a particular array, not a disk. To compute an array's usage ratio, DIVA examines each disk in the array and divides the sum of each disk's used space by the sum of each disk's total capacity.

Example:

An array composed of two disks of the same size, one 100% full and one 50% full, is considered by DIVA to be 75% full. The same usage ratios with one disk of 2 Terabyte and one disk of 1 Terabyte will result in an 83% filling ratio for the array; 100% of 2 TB and 50% of 1 TB yields a total of 2.5 TB used, divided by a 3 TB total capacity.

For more information on watermarking arrays, see *Watermarks and Arrays*.

General Watermarking Rules

Telestream recommends that you adhere to the following general watermarking rules:

Do not set the High Watermark too high.

When the HWM is reached, SPM begins deleting expired instances. This process may take some time, especially if the instances are small, because the purge will require

more DeleteInstance operations. Confirm that any archive activity will not store enough additional instances on the array during this process to fill it to 100%.

Example:

If the HWM is 90%, then the available space is 100 - 90 = 10%. If the array size is 2 Terabytes, this 10% represents 200 Gigabyte. If the SPM purge process encounters numerous small instances requiring deletion, the process will be slow and the archive activity may possibly store 200 Gigabytes (or more) during the purge's execution. This will fill the array to 100% during the process, and initiating Archive terminations.

Try setting the HWM to a lower value if this situation is experienced.

Do not set the Low Watermark too low.

The lower the LWM is set, the shorter period disk instances will be kept on disk. The shorter period minimizes the chance to restore from disk, and the benefit of having disk instances.

If the array contains persistent data that can't be purged (for example, objects belonging to a Storage Plan that keeps disk instances for an unlimited time, or user files not belonging to DIVA), the LWM must be set accordingly. If the persistent data accounts for 40% of the array's capacity, and the LWM is set to 30%, the purges will never complete. However, SPM will still purge what it can.

For more information on watermarking arrays, see *Watermarks and Arrays*.

Simple SPM Configuration Example

The following figure illustrates a simple SPM configuration.

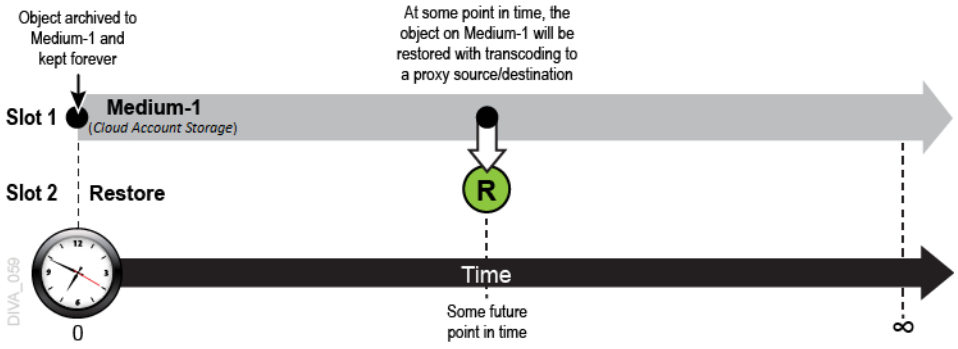
The following occurs in this scenario:

1. The object is archived to a watermarked array / tape group (Medium-1 with Watermark set to Yes) and held there for one day.
2. The object is immediately copied to DIVA Medium-2 (cloud storage), and stored there forever.
3. At the one day mark, the object is marked for deletion from the Medium-1 storage.

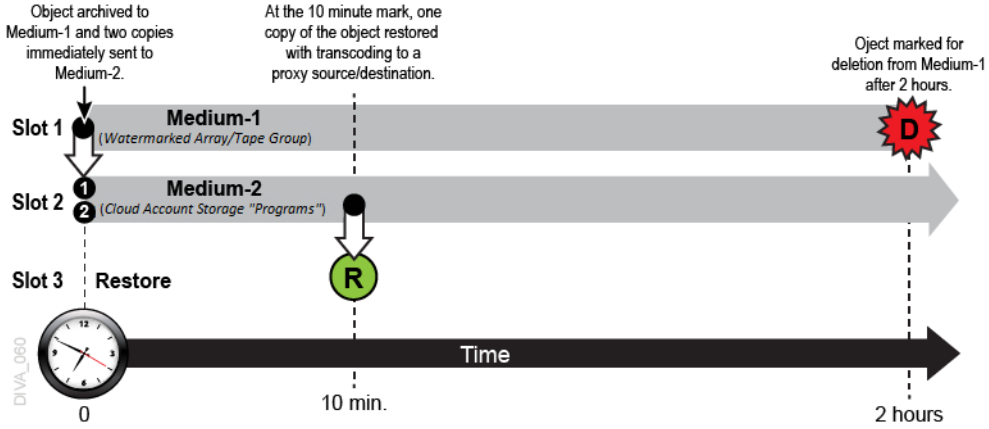
For more information about Slot Types and Slot configuration, see *SPM Slot Types*.

Additional SPM Configuration Examples

The following figure is an example of a more complex SPM configuration. In this example, one copy is kept on the SPM monitored cloud account forever. At some point in time, the object will be restored with transcoding to a proxy server.

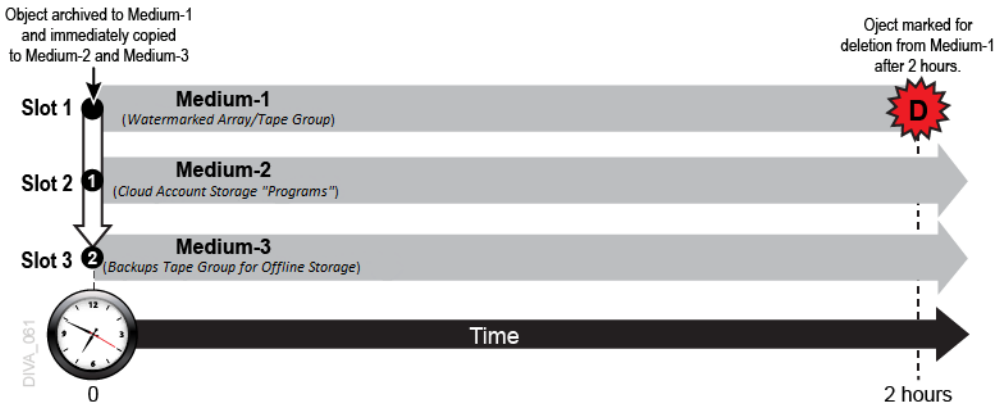


The following figure is an example of a more complex SPM configuration.



In this example, one copy is kept on the SPM watermarked array / tape group (with Watermark set to Yes) for two hours, and two copies are immediately made to a cloud account storage named Programs. At the ten minute mark, one copy will be restored to a proxy server.

This figure is an example of a SPM configuration with one copy left in the online tape group (Medium-2, named *Programs*), and a second backup copy made to a different tape group (Medium-3, named *Backups*) that is to be taken offline.



In this example, one copy is kept on the SPM watermarked array (with Watermark set to Yes) for two hours, and two copies are made immediately. One copy is made to a tape group named Programs, the second copy is made to a different tape group named Backups. The Backups tape group would typically be externalized from the DIVA system and sent to an Iron Mountain type of facility for offline storage.

Installation and Configuration

This chapter describes installation and configuration of SPM. References to installation and configuration procedures of additional software are included only to the extent required to allow interaction of those components with SPM. Refer to the specific additional software documentation in the DIVA Documentation Library on the DIVA Technical Support site for detailed information for full installation and configuration instructions for specific additional elements where necessary.

Installation Prerequisites

There are several prerequisites required before installing SPM as follows:

- The DIVA database must be Postgres 14 or later.
- The DIVA database user name and password must be the same ones used for the DIVA installation.
- DIVA must be installed and running.
- DIVA SPM Service must be installed and running.
- Users should have access rights to allow DSM to check monitored disk space.

SPM Configuration File Parameters

SPM uses a JSON file to store its configuration parameters. The configuration file, named SPM.settings.json.ini is supplied with SPM and is located in the `%DIVA_HOME%\Program\SPM\bin` folder.

Note: The file must be renamed to `SPM.settings.json` before use, and the parameter values must be changed to meet user requirements using a plain text editor (for example, Notepad or Notepad++).

Although all parameters are important, some are more important than others, and the following sections describe all parameters. Most parameters may be left at the installation defaults in the JSON file.

The following describes the parameters used in the SPM configuration file. Some parameters can also be set using the DIVA Web App. However, the settings in the SPM configuration file called *`ALLOW_OBJECT_DELETION`* and *`METADATA_ARCHIVE_TRANSFORMED_OBJECT_NAME`* always overrides any settings created through the DIVA Web App.

The following parameters are for Manager:

- The following are connection parameters for the DIVA database:
  - `User`

    This parameter identifies the DIVA database user name. There is no default; the typical user name is *diva*.
  - `Password`

    This parameter identifies the user's password. (Default: none).
  - `ConnectionString`

    This parameter identifies the DIVA database connection string. This is usually the SQL*Net address name declared in the local Postgres file for the DIVA Postgres instance. The typical connect string is lib5.
- The following are parameters for the SPM Monitor:
  - `DIVA_MANAGER_MONITOR_MAX_REQUESTS`

    SPM will check with Manager to find out how many jobs are currently processing before it sends any jobs. This parameter limits the number of jobs (processing and pending from all sources) the SPM service will attempt to send to Manager. If the number of jobs already processing on Manager is greater than this setting, SPM will not send any additional jobs at that time. The valid range is 10 to 500, and the default value is 250.
  - `DIVA_DELETE_MAX_REQUESTS`

    SPM will check with Manager to find out how many jobs are currently being processed. If the number is less than the `DIVA_MANAGER_MONITOR_MAX_REQUESTS` value, SPM will verify how many Delete Object and Delete Instance jobs are being processed. If the number is less than this parameter's value, SPM will send

additional Delete jobs equal to the difference between the number of Delete jobs processing, and the value of this parameter.

Example:

```
DIVA_MANAGER_MONITOR_MAX_REQUESTS = 100
DIVA_DELETE_MAX_REQUESTS = 50
```

In this example, when SPM checks with Manager, if DIVA is currently processing 75 jobs total, and only 25 of them are Delete jobs, SPM will send another 25 Delete jobs to DIVA, because both numbers are less that the parameter's value in the configuration file. The valid range is 1 to 300, and the default value is 300.

This value identifies how many of this type of job are simultaneously processed from the Action Queue.

This value can't exceed the value of `DIVA_MANAGER_MONITOR_MAX_REQUESTS`.

- `DIVA_RESTORE_MAX_REQUESTS`, `DIVA_TRANSCODE_ARCHIVE_MAX_REQUESTS`, and `DIVA_METADATA_ARCHIVE_MAX_REQUESTS`

  These parameters restrict the number of Restore, Transcode Archived and Metadata Archive jobs SPM can submit to the Manager simultaneously.

  These parameters must all be enabled or disabled together.

  The valid range is 0 to 300 and the default value is -1.

  The value used identifies how many of this type of job are allowed in the Action Queue. Using the default value of -1 disables these parameters. SPM will not execute these actions if the parameter is set to a value of 0.

  Example:

```
DIVA_RESTORE_MAX_REQUESTS=3
DIVA_TRANSCODE_ARCHIVE_MAX_REQUESTS=52
DIVA_METADATA_ARCHIVE_MAX_REQUESTS=100
```

  In this example, SPM will start normally using these configuration settings because they are all enabled, even though the values are different for each parameter (3, 52, and 100).

  Example:

```
DIVA_RESTORE_MAX_REQUESTS=3
DIVA_TRANSCODE_ARCHIVE_MAX_REQUESTS= -1
```

DIVA_METADATA_ARCHIVE_MAX_REQUESTS= -1

In this example, SPM will fail to start using these configuration settings because the first one is enabled with a value of 3, but the next two parameters are disabled. They all must be either enabled or disabled, not a combination of both.

- DIVA_MANAGER_MONITOR_DELAY

  This parameter identifies the number of seconds between checks of the number of jobs (processing and pending) on the Manager. The valid range is 1 to 600, and the default value is 30.

- DIVA_MANAGER_MONITOR_ACTION_DELAY

  This parameter identifies the number of seconds to delay an action for execution if DIVA is processing more than the value of the DIVA_MANAGER_MONITOR_MAX-_REQUESTS parameter during its execution. The valid range is 1 to 3600, and the default value is 60.

- The following are parameters for the SPM Controller:
  - ACTION_STEP_WATCH_DELAY

    This parameter identifies the minimum delay (in seconds) before the next check on the job status. The valid range is 1 to 60, and the default value is 10.

  - ACTION_QUEUE_SIZE

    This parameter identifies the maximum size of the Action Queue, including the number of actions that can be stored in the queue. The valid range is 1 to 300.

- The following are rest time parameters for SPM working tasks:
  - SPM_EXECUTION_THREAD_REST

    This parameter identifies the resting time for the Execution Task in seconds. The valid range is 0 to 600, and the default value is 5. The value of 0 should never be used and the minimum recommended setting is 1.

  - SPM_UPDATE_THREAD_REST

    This parameter identifies the resting time for the Update Task in seconds. The valid range is 0 to 600, and the default value is 20. Entering 0 disables the task completely.

  - SPM_LOAD_THREAD_REST

    This parameter identifies the resting time for the Load Task in seconds. The valid range is 0 to 600, and the default value is 20. Entering 0 disables the task completely.

Note: The Recovery Load Task will never rest; it stops upon completion of all work.

- The following are Row Chunk Size parameters that SPM will process simultaneously:
  - RECOVER_ACTIONS_CHUNK

    This parameter identifies the number of actions SPM will process (from the DIVA database) simultaneously during each call of the Recovery Task. The valid range is 1 to 65535, and the default value is 500.

- UPDATE_ACTIONS_CHUNK

  This parameter identifies the number of actions SPM will process simultaneously during each call of the Update Task. The valid range is 1 to 500, and the default value is 50.

- GET_ACTIONS_CHUNK

  This parameter identifies the number of actions SPM will process simultaneously during each call of the Load Task. The valid range is 1 to 100, and the default value is 25.

- UPDATE_ACTIONS_NEXT_UPDATE

  This parameter identifies the time (in minutes) added to the action's date of next update after it has been updated by the Update Task. DIVA SPM is more dynamic, and constantly reexamines actions. The date of next update is used internally to sort the actions to be reexamined. The valid range is 10 to 65535, and the default value is 20160 (two weeks).

- UPDATE_ACTIONS_RETRY_FAILED_DELAY

  This parameter identifies the delay (in minutes) before a long retry of a failed action. After this value is reached, the Update Task will reschedule all actions in the FAILED LONG state to be executed again. This parameter configures the retry of SPM actions and will not retry the action if its corresponding slot is modified after the action is put to a FAILED_LONG state—it will wait for the UPDATE_ACTIONS_NEXT_UPDATE to update the actions according to the latest slot changes before executing the action. The valid range is 10 to 65535, and the default value is 720 minutes.

- The following are SPM processing options:
  - ALLOW_OBJECT_DELETION

Caution: This parameter causes complete object deletion when enabled!

An object can be deleted when a Storage Plan has no slot at a given time. This feature must be managed and the Storage Plans defined carefully because many objects could potentially be lost according to the Storage Plan.

The valid settings are true or false, and the default value is false.

Set this field to true to enable object deletion. The Storage Plan definition in the DIVA database can either allow or deny deletion. If deletion is denied, configuring a Storage Plan to enable deletion will have no effect.

Note: The setting in the SPM configuration file for this parameter overrides any settings created through the DIVA Web App.

For example, if there is slot starting at day 0 and ending at day 30, and another slot starting at day 61 with an unlimited retention, objects are deleted after 30 days because there are no slots during the second month (days 31 to 60).

- DELETE_OBJECT_ONLY_LAST_INSTANCE

This parameter works with the ALLOW_OBJECT_DELETION parameter, and forces SPM to Delete Object at the end of the storage slot. Deletion only occurs if this object is the last instance in the entire DIVA system, and this last instance exists in the Storage Slot medium that has the highest Slot End Time among all Storage Slots of the Storage Plan. Otherwise, SPM will never perform a Delete Object at the end of the Storage Slot, it will always only do a Delete Instance instead. Refer to *SPM Delete Object Behavior* for more detailed information.

  - METADATA_ARCHIVE_TRANSFORMED_OBJECT_NAME

    This parameter defines whether SPM will use the original, or transformed, object name as the target object name in a METADATA_ARCHIVE action. This setting overrides any configuration performed through the DIVA Web App.

    The default setting is true. When this value is true, SPM uses the transformed object name. When this value is false, SPM uses the original object name.

- The following is a DSM task option:
  - DSM_DIW_REQUEST_PRIORITY

    This parameter identifies the priority of the jobs produced by DELETE INSTANCE actions. The valid range is 1 to 100, and the default value is 15.
- The following is a Rest Time parameter for DSM working tasks:
  - DSM_SPACE_MONITOR_THREAD_REST

    This parameter identifies the resting time for the DSM Task in seconds. The valid range is 0 to 600, and the default value is 10. Entering 0 disables the task completely.
- The following is a Row Chunk Size parameter that DSM will process simultaneously:
  - DSM_ACTIVATE_ACTIONS_CHUNK

    This parameter identifies the number of actions DSM will process at a given time during each call of the procedure. The valid range is 1 to 65535, and the default value is 50.
- The following is an SPM.settings.json Log level parameter:
  - LogLevel

    This parameter identifies the level of tracing to use for the logs. Refer to the default SPM configuration file (SPM.settings.json.ini) for more details.

Request Type Distribution

SPM contains three Request Type Distribution options that are defined in the SPM configuration file. These parameters restrict the number of Restore, Transcode Archived, and Metadata Archive actions that can be executed simultaneously. The default values for the following parameters are -1 in the SPM configuration file:

- DIVA_RESTORE_MAX_REQUESTS
- DIVA_TRANSCODE_ARCHIVED_MAX_REQUESTS
- DIVA_METADATA_ARCHIVE_MAX_REQUESTS

Acceptable values for these three parameters are -1, 0, and positive integers. Leaving the configuration file settings for these parameters to the default (-1) causes SPM to ignore the settings, and the defined behavior will not be changed.

Set all, or none, of these parameters to a value higher than the default (-1) setting. For example, a user can't set `DIVA_RESTORE_MAX_REQUESTS=3`, but keep `DIVA_TRANSCODE_ARCHIVED_MAX_REQUESTS= -1`. If only one or two parameters are defined with other values, SPM will refuse to start because of the incorrect configuration.

Note: Whenever these parameter values are changed, SPM must be restarted for the new values to take effect.

There are several rules for Request Type Distribution as follows:

- SPM must not execute more Restore jobs than the `DIVA_RESTORE_MAX_REQUESTS` value simultaneously. If the `DIVA_RESTORE_MAX_REQUESTS=0`, SPM will not execute any Restore jobs.
- SPM must not execute more Transcode Archived jobs than the `DIVA_TRANSCODE_ARCHIVED_MAX_REQUESTS` value simultaneously. If the `DIVA_TRANSCODE_ARCHIVED_MAX_REQUESTS=0`, SPM will not execute any Transcode Archived jobs.
- SPM must not execute more Metadata Archive jobs than the `DIVA_METADATA_ARCHIVE_MAX_REQUESTS` value simultaneously. If the `DIVA_METADATA_ARCHIVE_MAX_REQUESTS=0`, SPM will not execute any Metadata Archive jobs.
- SPM will attempt to reserve the configured number of actions in the SPM queue to execute the corresponding number of the jobs (if available). The number of simultaneous Restore jobs may be less than `DIVA_RESTORE_MAX_REQUESTS`, even if some Restore actions are ready to be executed immediately. This configuration must be observed for less than 50 percent of the execution time.

Example:

```
DIVA_RESTORE_MAX_REQUESTS=3

DIVA_MANAGER_MONITOR_MAX_REQUESTS=100
```

Workflow:

- If Restore Actions exist and are scheduled, SPM will execute three Restore jobs at the same time (most of the time).
- SPM will never execute more than three Restore jobs.
- Sometimes (less than 50 percent of operational time) SPM will execute 0, 1 or 2 Restore jobs when Restore Actions exist, and are scheduled.
- If no Restore Actions exist, and are scheduled, other actions will be executed so the total amount of jobs executed will be 100 (not 97).

By default, the Execute process will perform Delete Actions first, and then other job types by Slot Priority. If this feature is configured, SPM will still perform Delete Actions

first, and then decide the number of Copy, Restore, Metadata Archive, and Transcode Archived jobs to send, and then for each type of job, sort the actions by Slot Priority.

Example:

- Two Copy Slots have the defined priorities 90 and 95, and 1000 Copy Actions are to be executed.
- There are two Restore Slots with priorities 30 and 35, and 100 Restore Actions to be executed.
- `DIVA_RESTORE_MAX_REQUESTS=3`
- `DIVA_MANAGER_MONITOR_MAX_REQUESTS=100`

Workflow:

- SPM will disregard the priority between the slot types, so 97 Copy jobs will be scheduled, and 3 Restore jobs will be scheduled.
- SPM will schedule Restore jobs with the priority of 35 first.
- Delete Object and Delete Instance jobs always have higher priority.

Storage Plan Definition

SPM requires configuration of Storage Plans before execution. Configure the SPM Storage Plans using the DIVA Web App.

There are two tabs in the DIVA Web App related to SPM configuration; Storage Plans and Slots.

Storage Plan Settings

The Storage Plan Settings in DIVA has three sections:

- Storage Plans
- Media Groups
- Filters

Each of these sections is configured independently, but is related with topics to complete the configuration.

Recommended Best Practices

Telestream recommends using the following order when configuring SPM through the DIVA Web App:

1. Create the Storage Plans.
2. Create the Media Groups.
3. Create the Filters.
4. Create the slots (using the Slots Settings).

Creating the Storage Plan

The first step in configuring SPM is to create the Storage Plans. Use the following procedure to create a storage Plan:

1. Click the + in the Storage Plans section of the Storage Plan Settings in the DIVA Web App.
2. When the Add New Row dialog box opens, begin configuring the new Storage Plan. Enter an name in the new Storage Plan Name field.
3. Use the lists to select the values for the parameter fields as follows:
   - Storage Plan Name

     This parameter identifies the name of the Storage Plan.
   - Allow Last Instance Deletion

Caution: Use this parameter with caution. Data loss can occur if set incorrectly.

     This parameter determines whether SPM is allowed to delete the last instance of an object when there is only one instance remaining on the DIVA system. Set to `Y` to allow, or `N` to deny, deletion of the last instance of an object.

     This parameter only applies to Storage Slots, and is overridden by the configuration of the ALLOW_OBJECT_DELETION Parameter in the SPM configuration file.
   - Please specify origin (Internal/External)

     Set to `I` for internal, or `E` for external origins. This parameter is typically used for all SPM actions. The internal (I) setting is for objects contained within the DIVA system and is the typical selection. The external (E) setting is mainly used during data migration in combination with the SPM API. Only Telestream should use the external setting in this release. For example, for use with DIVA Automatic Data Migration from Avalon to DIVA.
   - Tape Group/Array Name

     Select the default media from the list of media managed in the DIVA system. DIVA uses the selected media for object placement when a new object is found, and the Storage Plan in use has an active Storage Slot. This placement occurs when an Archive Request is submitted to DIVA without a target Media, and only a Storage Plan selected in the job.
4. Click OK to save the Storage Plan. The new Storage Plan will be displayed in the Storage Plans section of DIVA Web App.

Creating Filters

Filters determine whether an action must be performed on an object, or if it should be ignored. Filters only function for mediums defined in the Mediums list and monitored by SPM.

When a new object is archived into DIVA, and it matches a defined filter, a Storage Plan is assigned to that object as follows:

- If the new object doesn't meet any filter criteria, the default Storage Plan will be assigned.
- If an object satisfies multiple filters, it will be assigned the Storage Plan of the matching filter with the lowest Filter ID.

To create a filter, do the following:

1. Navigate to the DIVA Web App Storage Plan settings.
2. Click the + icon on the top of the Filters section.
3. When the Add New Row dialog box appears, begin configuring the new filter.

   Leaving the default * in Object Collection, Object Source, and Object Name fields causes no filtering to be identified for these parameters.
4. Enter the information in the appropriate fields, and use the menu lists to select the remaining options as follows:
   - `Filter Name`

     Enter the Filter Name in the Filter Name field.
   - `Id`

     The Filter ID is auto-generated by SPM starting with Filter 0.
   - `Media Filter`

     Use the menu to select the Mediums previously created.
   - `Object Collection Filter`

     Enter any object collection filters required. Leaving the default * will cause no filtering for this parameter.
   - `Object Source Filter`

     Enter any object collection filters required. Leaving the default * will cause no filtering for this parameter.
   - `Object Name Filter`

     Enter any object collection filters required. Leaving the default * will cause no filtering for this parameter.
   - `Min. object size`

     Enter the minimum object size.
   - `Max. object size`

     Enter the maximum object size.
   - `Size units`

     Use the menu to select the size units (B, KB, MB, and GB). The initials refer to Byte, KiloByte, MegaByte, and GigaByte respectively.
   - `Storage Plan`

     Use the menu to select the Storage Plan to associate with this filter.
5. Click OK to complete the process.

   The new Filter is now displayed in the Filters section of the Storage Plans view.

Assigning Storage Plans to an Object

There are several alternate methods of assigning a Storage Plan to an object including the following:

- Assigning a new Storage Plan to an object using the DIVA Web App.
- Using the DIVA REST API.
- Including a Storage Plan name when submitting an archive job forcefully applies a Storage Plan to the object and avoid the filters.

Media Mapping enables configuring an alternate target media, and to forcefully apply a Storage Plan, avoiding filters, when an object is archived to a specific Cloud Bucket or disk array.

To assign a storage plan to an object, do the following:

1. In the DIVA Web App, browse to *Configuration* > *Resources* > *Cloud Storage* > *Cloud Buckets*, or *Configuration* > *Resources* > *Disk Storage* > *Arrays*.
2. To open the *Add new row* dialog, in the *Media Mapping* dialog, click the + icon.
3. Enter the appropriate information in the fields as follows:
   - `Id`

     This field is automatically system generated and is not editable.
   - `Name`

     Enter the name to use for this mapping in the Name field.
   - `From`

     Use the menu list to select the media to use for this mapping.
   - `Map to Media`

     Use the menu list to select the Destination Media for this mapping.
   - `Map to Storage Plan`

     Use the menu list to select the Storage Plan to use for this mapping.

For example:

- Media Array1 is mapped to Array2. In this scenario, objects archived to the media named Array1 will be transferred to the media named Array2 as its destination. If there is a filter configured with target media as Array2, the Storage Plan configured in that filter will be assigned to the archived object.
- Media Array1 is mapped to the Storage Plan named storagePlan1. In this scenario, objects archived to the media named Array1 will be assigned the Storage Plan named storagePlan1, and avoid filters.
- Media Array1 is mapped to Array2, and also the Storage Plan named storagePlan1. In this scenario, objects archived to the media named Array1 will be transferred to the media named Array2 as its destination, assigned the Storage Plan named storagePlan1, and avoid filters.

Slots

The following sections describe the process for creating each different type of slot.

Configuring Storage Slots

The Storage Slot is the only slot that has two actions associated with it. It will create a copy of the object when it starts and delete it when the slot ends, unless the medium used is watermarked.

For watermarking details, see *Watermark-Based Disk Cleaning Management*.

The copy action will occur only if the slot target medium doesn't already have the required number of instances identified in the slot configuration. The Delete Action will only occur if the slot is specifically configured to perform the Delete Action (slot has an ending time defined).

To configure a Storage Slot, do the following:

1. Open the DIVA Web App, and navigate to the *Configuration* > *Resources* > *Storage Plans* > *Slots* view.
2. Click the + icon on the top right of the Slots section.
3. When the Slot Configuration dialog box opens, configure the options as follows:
   - `Storage Plan`

     Use the menu list to select the Storage Plan to be associated with this slot.
   - `Request Type`

     Select the type of job for this slot (storage, transcode archived, metadata archive, or restore). Select STORAGE in this case.
   - `Slot Name`

     Enter a name for the slot in the Slot Name field.
   - `Slot Begin Time (Minutes)`

     This parameter identifies when the action will occur. If set to zero the action is initiated as soon as the object is archived. If set to another number, for example 10 (minutes), the action will execute 10 minutes after the object is archived.
   - `Slot End Time (Minutes)`

Caution: If the Slot Begin Time and Slot End Time are both set to zero, then the behavior is unknown, except for Storage Slots.

Setting this parameter to -1 indicates that the slot will never end (it is permanent). If set to another number, for example 10 (minutes), the Delete Action will execute 10 minutes after the object is archived.

If a Storage Slot's Slot Begin Time and Slot End Time are the same (for example, 0, 0 or 10, 10), the slot is what may be referred to as an idle slot. SPM creates a Copy and a DeleteInstance Action and these actions get updated but not executed.

The only exception is if the medium is watermarked. If the medium is watermarked then the DeleteInstance Action will be executed when a disk clean-up is necessary and it will go through the usual POSTPONED through COMPLETED cycle.

In practice, creating an idle slot is useful for mediums where an external tool creates instances (not SPM), but use of the SPM's watermark-based clean-up mechanism is desired.

Example (typical slot):

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): 20

Workflow:

The original object is archived onto the source medium.

Ten minutes later, the slot actions begin and it is copied to the target medium.

SPM deletes the copy from the target medium 10 minutes later.

Example (idle slot):

Slot End Time (Minutes): 0

Slot Begin Time (Minutes): 0

Workflow:

The object is archived on medium Array1, and the idle slot is defined for medium Array2. An Array2 Copy Action is created, and the status is set to SCHEDULED. Then, an Array2 DeleteInstance Action is created, and the status is set to COMPLETED.

An external tool creates an instance on Array2 using the Copy function. The Array2 Copy Action status is set to COMPLETED, and the Array2 DeleteInstance Action status is set to POSTPONED.

The Array2 hits the High Watermark. The Array2 DeleteInstance Action is executed, and the status is set to COMPLETED.

The Array2 Copy Action status remains as COMPLETED.

– Enabled

Determines whether the slot is active. Selecting Y indicates the slot is enabled, and N indicates it is disabled. When a slot is disabled its actions aren't executed, however they are still created, and the status is updated. When the slot is enabled again, pending actions are executed immediately.

Examples:

A new object that is archived and assigned to a Storage Plan with a Tape Storage Slot will have the Tape Slot's Copy Action created, but not executed. Therefore, no Copy job is sent to DIVA.

A DeleteInstance Action reaching its execution time will not send the Delete Instance job to DIVA, but if somebody manually deletes that instance using the DIVA Web App, the action will be updated to the COMPLETED status.

– Request Execution Begin Time (HH:MM)

  This field is the slot execution window opening time in 24 hour time. If set to 00:00 the slot opens at the beginning of each day. If set to another time, for example 08:00, it will open at that time (in this case 8:00 am).

– Request Execution End Time (HH:MM)

  This field is the slot execution window closing time in 24 hour time. If set to 24:00 the slot closes after each day. If set to another time, for example 17:00, it will close at that time (in this case 5:00 pm).

  Setting the Request Execution Begin Time and Request Execution End Time to something other than an entire 24 hour period enables scheduling of different functions through SPM at different times of the day (for example overnight Requests). If an object is archived to the source medium outside of the designated time frame for the slot, even though the object meets the Filter and Storage Plan criteria, no actions from this particular slot will occur.

  If the original object was archived to the source medium while the slot was inactive (closed) and has passed the Slot End Time, when the slot finally becomes active (open), no Delete Action will occur.

Note: This is not the case for Storage Slots. For Storage Slots the DeleteInstance Action will still be executed.

Example (non-Storage Slots):

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): 20

Request Execution Begin Time (HH:MM): 08:00

Request Execution End Time (HH:MM): 14:00

Workflow:

A new object is placed onto the source medium at 07:00 (while the slot is closed). If the slot had been open at 07:00 when the new object was created it would be copied to the target at 07:10 and then deleted from the target at 07:20. Because the slot was not open when the new object was created, and the Slot End Time has passed when this slot opens at 08:00, no actions will be taken on the object.

If a new object is created on the source medium at 14:15 today, then tomorrow when this slot opens, no actions will be taken on the object because the Slot End Time has passed.

If an object is created on the source medium at 07:59 (before the slot opens), at 08:09 (10 minutes after the object existed on the source) the object will be copied to the target medium because the Slot End Time has not yet passed. Then at

08:19 (20 minutes after the object existed on the source) the object instance on the target will be deleted.

Example (Storage Slots):

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): 20

Request Execution Begin Time (HH:MM): 08:00

Request Execution End Time (HH:MM): 14:00

Workflow:

A new object is placed onto the source medium while the slot is closed. A Copy Action is created and the status is set to COMPLETED. A DeleteInstance Action is created and the status is set to SCHEDULED. No actions are executed because the slot is closed.

The disk instance expires after 20 minutes.

The slot opens the next day. The DeleteInstance Action is executed, and the status is set to COMPLETED. The Copy Action status remains as COMPLETED.

– Job priority

  This field identifies the priority of the job and order of execution when the action is submitted to the Manager.

– Media Associated with Slot

  Select the target medium to associate with the slot from the list.

– Req. Number of Instances

  The number of object instances that should exist on the target medium. If the actual number of instances is less than this setting, additional copies will be created. This parameter is only applicable to Storage Slots.

  Example:

  Req. Number of Instances: 2

  Workflow:

  One object instance already exists on the target medium.

  Because there is only one instance existing, and this parameter is set to 2, another copy will be made to another disk in the array, or to another tape in the tape group.

– Once Only

  Determines whether the slot can run multiple times. Setting this to N indicates it can run multiple times, while setting this to Y indicates it will only run once.

Note: This parameter is only applicable for Storage Slots and must be set to Y when using Mixed Mode Watermarking.

Example:

Once Only: N

Workflow:

An object was copied to the target medium, but somebody manually deleted the copied instance. This slot will realize the object doesn't exist on the target medium and run again, resulting in another copy on the target medium being created.

Example:

Once Only: Y

Workflow:

An object was copied to the target medium, but somebody manually deleted the copied instance. This slot will not run again, resulting in no object instance on the target medium (because it had been manually deleted).

If the Slot End Time for deletion has been set, when the Slot End Time is reached, SPM will automatically mark the object instance on the target medium as being deleted because it was manually deleted before the Slot End Time was reached.

Click OK to complete the process. The new slot will be displayed on the Slots view in the DIVA Web App.

Configuring Nearline Storage Slots

Two additional Quality of Service (QOS) parameters—NEARLINE_ONLY and NEARLINE_AND_DIRECT—can be specified via the API or the DIVA Web App in Restore and Multiple Restore jobs. The default QOS for a Restore or Multiple Restore job is NEARLINE_AND_DIRECT. The default QOS for all other jobs remains DIRECT_AND_CACHE.

When DIVA receives either of these QOS values, it will initiate a DIRECT restore from any available disk instance, regardless of whether the disk instance is located on a disk with an actor-disk connection that is configured for Nearline. If no disk instance is found, DIVA will not terminate the job; instead it creates a disk instance and proceeds in the following sequence:

1. Check for NEARLINE disks during disk selection.
2. If no disks are available, then for QOS:
   - `NEARLINE_ONLY`

     The workflow terminates and an error is generatedL "No AVAILABLE Actor-Disk connection is configured with Nearline storage".
   - `NEARLINE_AND_DIRECT`

     `DIRECT RESTORE` will be used.
3. Reserve space for permanent storage during the restore.

4. Create a permanent disk instance in memory.

   If the disk instance can't be created, then for QOS:

   – `NEARLINE_ONLY`

     The workflow will terminate.

   – `NEARLINE_AND_DIRECT`

     `DIRECT RESTORE` will be used.

   If the disk instance can be created, the workflow will proceed to the next step.
5. Write the disk instance to the DIVA database upon success or failure of the Restore.
6. Save component checksums for the new disk instance.
7. Update storage capacity.
8. Update last access date of object instance, if not already updated.

If all disk instances are busy, a delayed solution will be generated.

DIVA will terminate a Restore Instance job with a QOS of `NEARLINE_ONLY`, or a QOS of `NEARLINE_AND_DIRECT`, by sending an error message to the DIVA Web App stating Nearline and Nearline & Direct QOS aren't supported for Restore Instance, and sending an Invalid Parameter error code to the API.

Nearline storage is defined using the same technique for defining the disk use. The Nearline type is defined by specifying one of two uses under Actor-Disk Connections called `STORAGE_AND_NEARLINE` and `CACHE_AND_STORAGE_AND_NEARLINE`. Any disk defined with one of these uses can be used for any of these operations.

An SPM Storage Slot supports this workflow for purging disk instances created during Restore jobs.

- If the disk instances are created on the same medium used in the Storage Slot, a Delete Instance action in the Storage Slot will be rescheduled.
- If the disk instance is created on a different medium, the Storage Slot's Slot Begin Time and Slot End Time must be set to 0 for the associated medium.

SPM supports purging of disk instances created by DIVA during Restore jobs in the following two work flows:

1. Nearline Instances are stored on the same medium, which is also used in an SPM Storage slot.
   – In this work flow it is assumed that the medium used for Nearline Storage must also be used in a SPM Storage Slot.
   – The Storage Slot must have Once Only set to N.
   – Nearline instances aren't copied until the delete instance action is completed for the Storage Slot on the medium.
   – The medium used must be watermarked.
   – The disk cleaning strategy for the medium must be based on the Object Last Access Time.

- After the action is put into the SCHEDULED state, SPM will start processing and put it into the POSTPONED state until the disk reaches its High Watermark.
- After the disk reaches its High Watermark value, all of the Delete Instance Actions in the POSTPONED state will be executed based on the Disk Cleaning Strategy configured for the medium. In this case it will be the Object Last Access Time.

2. A separate medium is used for Nearline Storage.

- In this workflow a Storage Slot is configured on the medium with the Slot Begin Time and Slot End Time is set to 0.
- The Storage Slot must have Once Only set to N.
- Setting the Slot Begin Time and Slot End Time to 0 causes SPM to generate a Storage Slot Action for this medium. However, it will be in the COMPLETED state by default.
- The medium used must be watermarked.
- The Disk Cleaning Strategy for the medium must be based on Object Last Access Time.
- After the action is put into the SCHEDULED state, SPM will start processing and put it in the POSTPONED state until the disk reaches its High Watermark.
- After the disk reaches its High Watermark value, all of the Delete Instance Actions in the POSTPONED state will be executed based on the Disk Cleaning Strategy configured for the medium. In this case it will be the Object Last Access Time.

A medium must not be configured for only one of previously mentioned workflows. Every medium that has a disk used for STORAGE_AND_NEARLINE or CACHE_AND_STORAGE_AND_NEARLINE must be configured with any one of the mentioned workflows for SPM to purge Nearline instances.

Configuring Transcode Archived Slots

Transcode archived slots require configuration on the *SLOTS* view and contain only a single action. This slot will take an existing clip in a DIVA Object, transcode it to another format, and archive the resulting clip as a new object.

Because a transcode archived job creates a new object in DIVA, it requires directions as how to name the new object (Transformation Rules).

Note: Each actor defined as a Transcoder must also be defined as a LOCAL type server.

To configure a transcode archived slot, do the following:

1. Launch the DIVA Web App and navigate to the *Configuration* > *Resources* > *Storage Plans* > *Slots* view.
2. Click the + icon to the right of the *SLOTS* heading.

   DIVA opens the *SLOTS* view.
3. Select *Transcode Archived* from the *Job Type* list.

4. Enter the following field values:
   - *Name*

     Enter a name for the slot in the *Name* field.
   - *SPM*

     Select the storage plan to associate with this slot from the drop-down menu.
   - *Job Type*

     Select the type of job for this slot (storage, transcode archived, metadata archive, or restore). Select `Transcode Archived` in this case.
   - *Slot Begin Time (Minutes)*

     This parameter identifies when the action will occur. If set to zero the action is initiated as soon as the object is archived. If set to another number, for example 10 (minutes), the action will execute 10 minutes after the object is archived.
   - *Slot End Time (Minutes)*

Caution: If the Slot Begin Time and Slot End Time are both set to zero, then the behavior is unknown, except for Storage Slots.

Setting this parameter to -1 indicates the slot will never end (it is permanent) and the Transcode Archived Action will occur regardless of whether the slot is open or closed.

Example:

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): -1

Workflow:

The original object is archived onto the source medium. Ten minutes later, the slot actions begin and it is transcoded onto the target medium.

Because the Slot End Time is set to -1, this slot will never end. If the slot is called during a period when the slot is closed, then as soon as the slot is open the transcode will take place. If this was set to another number (other than -1), the normal rules will apply as described later in the Request Execution End Time parameter.

- *Enabled*

  Determines whether the slot is active. To enable the slot, drag the slider to the right. To disable the slot, drag the slider to the left. When a slot is disabled its actions aren't executed, however they are still created, and the status is updated. When the slot is enabled again, pending actions are executed immediately.
- *Job Execution Begin Time (HH:MM)*

  This field is the slot execution window opening time in 24 hour time. If set to 00:00 the slot opens at the beginning of each day. If set to another time, for example 08:00, it will open at that time (in this case 8:00 am).
- *Job Execution End Time (HH:MM)*

This field is the slot execution window closing time in 24 hour time. If set to 24:00 the slot closes after each day. If set to another time, for example 17:00, it will close at that time (in this case 5:00 pm).

Setting the Request Execution Begin Time and Request Execution End Time to something other than an entire 24 hour period enables scheduling of different functions through SPM at different times of the day (for example overnight Requests). If an object is archived to the source medium outside of the designated time frame for the slot, even though the object meets the Filter and Storage Plan criteria, no actions from this particular slot will occur.

If the original object was archived to the source medium while the slot was inactive (closed) and has passed the Slot End Time, when the slot finally becomes active (open), the action taken (or not taken) is determined by the Slot End Time setting.

Example:

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): 20

Request Execution Begin Time (HH:MM): 08:00

Request Execution End Time (HH:MM): 14:00

Workflow:

A new object is placed onto the source medium at 07:00 (while the slot is closed). If the slot had been open at 07:00 when the new object was created it would be transcoded to the target at 07:10. However, because the slot was not open when the new object was created, and the Slot End Time has passed, when this slot opens at 08:00, no action will be taken on the object.

If a new object is created on the source medium at 14:15 today, then tomorrow when this slot opens, no actions will be taken on the object because the Slot End Time has passed.

If an object is created on the source medium at 07:59 (before the slot opens), at 08:09 (10 minutes after the object existed on the source) the object will be transcoded to the target medium because the Slot End Time has not yet passed.

Example:

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): -1

Request Execution Begin Time (HH:MM): 08:00

Request Execution End Time (HH:MM): 14:00

Workflow:

A new object is placed onto the source medium while the slot is closed. If the slot had been open at 07:00 when the new object was created, it would have been transcoded to the target at 07:10.

Because the slot was not open when the new object was created, and the Slot End Time is set to never end (-1), when this slot opens at 08:00, the Transcode Action will be completed on the object because the slot never ends.

If a new object is created on the source medium at 14:15 today, then tomorrow when this slot opens, the Transcode Action will be completed on the object because the slot never ends.

– *Request priority*

This field identifies the priority of the job and order of execution when the action is submitted to DIVA.

– *Medium Associated with Slot*

Select the target medium to associate with the slot from the list.

– *Req. Number of Instances*

The number of object instances that should exist on the target medium. If the actual number of instances is less than this setting, additional copies will be created. **This parameter is only applicable to Copy Actions in Storage Slots.**

– *Once Only*

This parameter is only applicable for Storage Slots and must be set to Y when using Mixed Mode Watermarking.

5. Click OK to complete the process. The new slot is displayed on the Slots view under the *Configuration* > *Resources* > *Storage Plans* > *Slots* view.

The set of toolbar at the bottom of the Slot Configuration view displays these buttons:

– Restore, Transcode & Metadata Archive

– Storage

The Transcode Archived Slot uses the Restore, Transcode & Metadata Archive to configure additional parameters including:

```
– Quality Of Service
– Request Options
– Request Comments
– Object Name Filter
– Object Source Filter
– Object Collection Filter
– Cascade Delete Links
– Transformation Rules (for renaming)
```

The object filters allow restricting the Transcode process to particular objects. Use them if all objects belonging to a Storage Plan do not need to be transcoded.

The transformation rules are rules associated with the object filters to transform the object name and collection for transcode archived and metadata archived.

6. Using the Restore, Transcode & Metadata Archive fields on the Slot Configuration view, configure the Object Filters and Transformation Rules as required:
   - **Object Name Filter**—Enter the Object Name filtering criteria to use for this filter. An asterisk is a wildcard character and the default entry in this field.
   - **Object Source Filter**—Enter the Object Source filtering criteria to use for this filter. An asterisk is a wildcard character and the default entry in this field.
   - **Object Collection Filter**—Enter the object collection filtering criteria to use for this filter. An asterisk is a wildcard character and the default entry in this field.
   - **Target Name Template**—Enter the name for the transcoded clip on the target medium. The caret (^) character indicates the system should use the original name.
   - **Target Collection Template**—Enter the Collection for the transcoded clip on the target. The caret (^) character indicates the system should use the original name.

     If the object matches the Object Filter criteria, it will continue to be processed using the Transcode_Archived Slot. If the filter criteria aren't matched then no further action is taken on the object for this slot.

     Transformation Rules transform the Object's Name and Collection during the job, and the Transformation Name and Transformation Collection need to be added. At least one of these two values should be changed from the default.

     Using the caret (^) in the Target Name Template, or Target Collection Template, or both, will cause SPM to use the original clip name, or the original clip Collection, or both. The caret can be used with additional naming conventions (for example, ^_wm9 or ^_cat).

     Using only the caret with no additional characters causes the system to initiate an error stating that an object with that name, or collection, or both, already exists. Changing just one of these parameters avoids this error and creates the new transcoded clip on the target.
7. Similar to the Transformation Rules, the Request Templates parameters are located at the bottom of the Slot Configuration view. Configure the additional parameters as required:
   - **Object Unmanaged Storage Repository**—This field is not accessible because the newly created object is always archived from the actor that performed the transcoding.
   - **Quality Of Service**—Select the appropriate Quality Of Service from the list for this job. The Quality Of Service parameter has seven different levels:

* *DEFAULT*—Archiving is performed according to the default Quality Of Service (`DIRECT&CACHE` for archive operations).
* *CACHE-ONLY*—Use cache archive only.
* *DIRECT-ONLY*—Use direct archive only.
* *CACHE&DIRECT*—Cached transfers from, or to, a server to, or from, DIVA are preferred, but direct transfers will occur if it is not possible to process the job in cache mode.
* *DIRECT&CACHE*—Direct transfers from, or to, a server to, or from, DIVA are preferred, but cache transfers will occur if it is not possible to process the job in direct mode.
* *NEARLINE-ONLY*—If a Nearline disk instance exists, the data is transferred from Nearline disk to the destination server. Alternatively, the data is first transferred entirely to Nearline storage on disk from tape, and then transferred to the destination server. If no Nearline service is available, the job will be terminated.
* *NEARLINE&DIRECT*—If Nearline transfer is not available (for example, no actor with Nearline storage is available), a direct transfer is performed instead.

- `Request Options`—`Enter the transcoding options in this field based on the transcoders defined in the DIVA Web App. The available options are as follows:`
  * `-tr_names`
    This option specifies the transcoder to use as defined in the DIVA Web App Transcoders section.
  * `-tr_archive_format`
    This option specifies the factory and format to be transcoded to for Archive Slots.
  * `-tr_restore_format`
    This option specifies the factory and format to be transcoded to for Restore Slots.
- `Request Comments`

  Enter any comments in this field to be sent to DIVA with this action job.
- `Pathroot`

  This field is not accessible because the File Path Root for the archive of the transcoded object is generated internally by the transcode archived job.
- `Cascade Delete Links`

  This parameter determines whether to perform a Delete Action on just the original object, or on both the original and the transcoded object. If set to Y, when the original is deleted, the transcoded copy will also be deleted. When set to N, when the original is deleted, the copy remains untouched.
- `Cascade Restore Links`

  This parameter determines whether to perform a Restore Action on just the original object, or on both the original and the transcoded object. If set to Y, when the original is restored, the transcoded copy will also be restored. When set to N, when the original is restored, the copy remains untouched.

Configuring Metadata Archive Slots

Configure metadata archive slots on the *SLOTS* view. These slots contain only a single action. A slot archives a metadata file after an object is archived. The metadata file's filename must be the object name with no extension, and must reside on a server and path that are defined in the *ADD SLOT* view. The object name and object collection used for the metadata file archive are controlled by the Transformation Rule settings.

To configure metadata archive slot, do the following:

1. Launch the DIVA Web App, and navigate to *Configuration* > *Resources* > *Storage Plans* > *Slots*.
2. Click the + icon to the right of the *SLOTS* title.

   DIVA opens the *ADD SLOT* view.
3. Enter field values as follows:
   - *Name*

     Enter a name for the slot.
   - *SPM*

     Select the type of storage plan to be associated with this slot from the drop-down menu.
   - *Job Type*

     Select the type of job for this slot from the drop-down menu. Select *Metadata Archive* in this case.
   - *Slot Begin Time (Minutes)*

     This parameter identifies when the action occurs. If set to zero the action is initiated as soon as the object is archived. If set to another number, for example 10 (minutes), the action executes 10 minutes after the object is archived.
   - *Slot End Time (Minutes)*

Caution: If the Slot Begin Time and Slot End Time are both set to zero, then the behavior is unknown, except for Storage Slots.

Setting this parameter to -1 indicates that the slot never ends (it is permanent), and the Metadata Archive action occurs whether the slot is open or closed.

Example:

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): -1

Workflow:

The original object is archived onto the source medium.

Ten minutes later, the slot actions begin and the metadata file is archived onto the target medium.

Because the Slot End Time is set to -1, this slot never ends. If the slot is called during a period when the slot is closed, as soon as the slot is open the archive

takes place. If this was set to a number other than -1, the normal rules apply as described later in the Request Execution End Time description.

– *Job Execution Begin Time (HH:MM)*

  This field is the slot execution window opening time in 24-hour time. If set to 00:00 the slot opens at the beginning of each day. If set to another time, for example 08:00, it will open at that time (in this case 8:00 am).

– *Job Execution End Time (HH:MM)*

  This field is the slot execution window closing time in 24-hour time. If set to 24:00 the slot closes after each day. If set to another time, for example 17:00, it closes at that time (in this case 5:00 pm).

  Setting the *Job Execution Begin Time* and *Job Execution End Time* to something other than an entire 24-hour period enables scheduling of different functions through SPM at different times of the day (for example overnight jobs). If an object is archived to the source medium outside of the designated time frame for the slot, even though the object meets the Filter and Storage Plan criteria, actions taken on the object are determined by the setting of the *Slot End Time* parameter.

  If the original object was archived to the source medium while the slot was inactive (closed) and has passed the Slot End Time, when the slot finally becomes active (open), the action taken (or not taken) is determined by the Slot End Time setting.

  Example:

  Slot Begin Time (Minutes): 10

  Slot End Time (Minutes): 20

  Request Execution Begin Time (HH:MM): 08:00

  Request Execution End Time (HH:MM): 14:00

  Workflow:

  A new object is placed onto the source medium at 07:00 (while the slot is closed). If the slot had been open at 07:00 when the new object was created the metadata file would have been archived to the target at 07:10. But because the slot was not open when the new object was created, and the Slot End Time has passed, when this slot opens at 08:00, no actions will be taken on the object.

  If a new object is created on the source medium at 14:15 today, then tomorrow when this slot opens, no actions will be taken on the object because the Slot End Time has passed.

  If an object is created on the source medium at 07:59 (before the slot opens), at 08:09 (10 minutes after the object existed on the source) the object will be copied to the target medium because the Slot End Time has not yet passed. Then at 08:19 (20 minutes after the object existed on the source) the metadata file will be archived to the target medium because the Slot End Time has not yet passed.

  Example:

  Slot Begin Time (Minutes): 10

Slot End Time (Minutes): -1

Request Execution Begin Time (HH:MM): 08:00

Request Execution End Time (HH:MM): 14:00

Workflow:

A new object is placed onto the source medium at 07:00 (while the slot is closed). If the slot had been open at 07:00 when the new object was created, the meta-data file would have been archived to the target at 07:10.

Because the slot was not open when the new object was created, and the Slot End Time is set to never end (-1), when this slot opens at 08:00, the Metadata Archive action will be completed on the object because the slot never ends.

If a new object is created on the source medium at 14:15 today, then tomorrow when this slot opens, the Metadata Archive action will be completed on the object because the slot never ends.

– *Media Associated with Slot*

Select the type of target media to associate with the slot from thedrop-down menu.

– *Job priority*

This field identifies the priority of the job and order of execution when the action is submitted to the Manager.

– *Enabled*

Determines whether the slot is active. To enable the slot, drag the slider to the right. To disable the slot, drag the slider to the left. When a slot is disabled its actions aren't executed. However they still are created when a new object is archived.

4. Similar to the Transformation Rules, the Request Templates parameters are located at the bottom of the Slot Configuration view in the Restore, Transcode & Metadata Archive tab.

Object Filters allow restricting the Metadata Archive process to particular objects. They can be used if archiving metadata files for all objects belonging to a Storage Plan is not desired.

Transformation Rules are rules associated with the object Filters to transform the name and Collection during the job.

If the object matches the Object Filter criteria it will continue to be processed using the Metadata_Archive job type. If the filter criteria aren't matched then no further action is taken on the object for this slot.

Transformation Rules transform the Object's Name and Collection during the job, and the Transformation Name and Transformation Collection must be added. At least one of these two values should be changed from the default.

Using the caret (^) in the Target Name Template, or Target Collection Template, or both, will cause SPM to use the original clip name, or the original clip Collection, or both. You can use the caret with additional naming conventions (for example, ^_wm9 or ^_cat).

Using only the caret with no additional characters causes the DIVA system to initiate an error stating that an object with that name, or Collection, or both, already exists. Changing just one of these parameters avoids this error and creates the new transcoded clip on the target.

5. Configure the additional parameters as required:

   - `Object Unmanaged Storage Repository`

     This field is not accessible because the newly created object is always archived from the actor that performed the transcoding.

   - `Quality Of Service`

     Select the appropriate Quality Of Service from the list for this job. The Quality Of Service parameter has seven different levels:

     * `DEFAULT`

       Archiving is performed according to the default Quality Of Service (DIRECT&CACHE for archive operations).

     * `CACHE-ONLY`

       Use cache archive only.

     * `DIRECT-ONLY`

       Use direct archive only.

     * `CACHE&DIRECT`

       Cached transfers from, or to, a server to, or from, DIVA are preferred, but direct transfers will occur if it is not possible to process the job in cache mode.

     * `DIRECT&CACHE`

       Direct transfers from, or to, a server to, or from, DIVA are preferred, but cache transfers will occur if it is not possible to process the job in direct mode.

     * `NEARLINE-ONLY`

       If a Nearline disk instance exists, the data is transferred from Nearline disk to the destination server. Alternatively, the data is first transferred entirely to Nearline storage on disk from tape, and then transferred to the destination server. If no Nearline service is available, the job will be terminated.

     * `NEARLINE&DIRECT`

       If Nearline transfer is not available (for example, no actor with Nearline storage is available), a direct transfer is performed instead.

   - `Request Options`

     This is only used if a user name and password are required for the server, otherwise leave this blank.

   - `Request Comments`

     Enter any comments in this field to be sent to DIVA with this action job.

   - `Pathroot`

     Enter the path to the metadata files on the server.

   - `Object Name Filter`

Enter the filtering criteria to use. An asterisk is a wildcard character and the default entry in this field.

- `Object Source Filter`

  Enter the Object Source filtering criteria to use for this filter. An asterisk is a wildcard character and the default entry in this field.

- `Object Collection Filter`

  Enter the object collection filtering criteria to use for this filter. An asterisk is a wildcard character and the default entry in this field.

- `Cascade Delete Links`

  This parameter determines whether to perform a Delete Action on just the original object or on both the original and the metadata object. If set to Y, when the original is deleted, the metadata object will also be deleted. When set to N, when the original is deleted, the metadata object remains untouched.

- `Cascade Restore Links`

  This parameter determines whether to perform a Restore Action on just the original object, or on both the original and the metadata object. If set to Y, when the original is restored, the metadata object will also be restored. When set to N, when the original is restored the metadata object remains untouched.

- `Target Name Template`

  For a metadata archive job, this parameter serves two purposes; it notifies SPM to look for an object with this parameter value on the server, and it notifies SPM to create a Metadata Archive file on the target using this parameter value. The caret (^) character indicates the system should use the original name.

- `Target Collection Template`

  Enter the collection for the transcoded clip on the target. The caret (^) character indicates the system should use the original collection.

6. Click OK to complete the process. The new slot will be displayed on the Slots view under the *Configuration* > *Resources* > *Storage Plans* > *Slots* view.

Configuring Restore Slots

Configure Restore Slots on the Slots view of the DIVA Web App. Restore Slots only contain a single action. This slot will take an existing DIVA object and restore it onto the designated server.

To configure Restore Slots, do the following:

1. Open the DIVA Web App, and navigate to the *Configuration* > *Resources* > *Storage Plans* > *Slots* view.
2. Click the + icon to the right of the *SLOTS* heading.

3. When the Slot Configuration dialog box opens, configure the options as follows:
   - *Name*

     Enter a name for the slot in the Name field.
   - *SPM*

     Select the Storage Plan to associate with this slot from the drop-down menu.
   - *Job Type*

     Select the type of job for this slot (storage | transcode archived | metadata archive | restore). Select RESTORE in this case.
   - *Slot Begin Time (Minutes)*

     This parameter identifies when the action will occur. If set to zero the action is initiated as soon as the object is archived. If set to another number, for example 10 (minutes), the action will execute 10 minutes after the object is archived.
   - *Slot End Time (Minutes)*

Caution: If the Slot Begin Time and Slot End Time are both set to zero, then the behavior is unknown, except for Storage Slots.

Setting this parameter to -1 indicates that the slot will never end (it is permanent), and the Restore Action will occur regardless of whether the slot is open or closed.

Example:

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): -1

Workflow:

The object is found on the mediums monitored by SPM.

Ten minutes later, the slot actions begin and the object is restored to the target server.

If the Slot End Time was greater than zero, and if the Restore Action has not executed before the Slot End Time is reached (because the slot remained closed after the Slot End Time was reached) then the Restore Action will never be executed. If this was set to another number (other than -1), the normal rules will apply as described later in the Request Execution End Time description.

- *Enabled*

  Determines whether the slot is active. Selecting Y indicates the slot is enabled, and N indicates it is disabled. When a slot is disabled its actions aren't executed. However, they are still created when a new object is archived.
- *Job Execution Begin Time (HH:MM)*

  This field is the slot execution window opening time in 24 hour format. If set to 00:00 the slot opens at the beginning of each day. If set to another time, for example 08:00, it will open at that time (in this case 8:00 am).
- *Job Execution End Time (HH:MM)*

This field is the slot execution window closing time in 24 hour format. If set to 24:00 the slot closes after each day. If set to another time, for example 17:00, it will close at that time (in this case 5:00 pm).

Setting the Request Execution Begin Time and Request Execution End Time to something other than an entire 24 hour period enables scheduling of different functions through SPM at different times of the day (for example overnight jobs). If an object is archived to the source medium outside of the designated time frame for the slot, even though the object meets the Filter and Storage Plan criteria, actions taken on the object are determined by the setting of the Slot End Time parameter.

If the original object was archived to the medium while the slot was inactive (closed) and has passed the Slot End Time, when the slot finally becomes active (open), the action taken (or not taken) is determined by the Slot End Time setting.

Example:

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): 20

Request Execution Begin Time (HH:MM): 08:00

Request Execution End Time (HH:MM): 14:00

Workflow:

A new object is placed onto the source medium at 07:00 (while the slot is closed). Because the slot was not open when the new object was created, and the Slot End Time has passed, (20 minutes after the object was placed on the medium, that is 7:20) when this slot opens at 08:00, no actions will be taken on the object.

Note: If the slot had been opened at 07:00, then when the new object was created it would have been restored to the target at 07:10.

If a new object is created on the source medium at 14:15 today, then tomorrow when this slot opens, no actions will be taken on the object because the Slot End Time has passed.

If an object is created on the source medium at 07:59 (before the slot opens), at 08:09 (10 minutes after the object existed on the source) the object will be restored to the server because the Slot End Time has not yet passed.

Example:

Slot Begin Time (Minutes): 10

Slot End Time (Minutes): -1

Request Execution Begin Time (HH:MM): 08:00

Request Execution End Time (HH:MM): 14:00

Workflow:

A new object is placed onto the source medium at 07:00 (while the slot is closed). Because the slot is not open when the new object was created, and the Slot End

Time is set to never end (-1), when this slot opens at 08:00, the Restore Action will be completed on the object because the slot never ends.

Note: If the slot had been open at 07:00 when the new object was created, it would have been restored to the server at 07:10.

If a new object is created on the source medium at 14:15 today, then tomorrow when this slot opens, the Restore Action will be completed on the object because the slot never ends.

– *Job priority*

This field identifies the priority of the job and order of execution when the action is submitted to the Manager.

4. Similar to the Transformation Rules, the Request Templates parameters are located at the bottom of the Slot Configuration view in the Restore, Transcode & Metadata Archive tab.

Object filters allow restricting the Metadata Archive process to particular objects. They can be used if archiving metadata files for all objects belonging to a storage plan is not desired.

Transformation Rules are rules associated with the object Filters to transform the Name and Collection during the job.

If the object matches the Object Filter criteria it will continue to be processed using the Metadata_Archive job type. If the filter criteria aren't matched then no further action is taken on the object for this slot.

Transformation Rules transform the object's name and Collection during the job, and the Transformation Name and Transformation Collection must be added. At least one of these two values should be changed from the default.

Using the caret (^) in the Target Name Template, or Target Collection Template, or both, will cause SPM to use the original clip name, or the original clip Collection, or both. The caret can be used with additional naming conventions (for example, ^_wm9 or ^_cat).

Using only the caret with no additional characters causes the DIVA system to initiate an error stating that an object with that Name, or Collection, or both, already exists. Changing just one of these parameters avoids this error and creates the new transcoded clip on the target.

5. Configure the additional parameters as required:

– `Object Unmanaged Storage Repository`

This field is not accessible because the newly created object is always archived from the actor that performed the transcoding.

– `Quality Of Service`

Select the appropriate Quality Of Service from the list for this job. The Quality Of Service parameter has seven different levels:

  * DEFAULT
    Archiving is performed according to the default Quality Of Service setting (*Direct And Cache* for archive operations).
  * CACHE-ONLY
    Use cache archive only.
  * DIRECT-ONLY
    Use direct archive only.
  * CACHE&DIRECT
    Cached transfers from, or to, a server to, or from, DIVA are preferred, but direct transfers will occur if it is not possible to process the job in cache mode.
  * DIRECT&CACHE
    Direct transfers from, or to, a server to, or from, DIVA are preferred, but cache transfers will occur if it is not possible to process the job in direct mode.
  * NEARLINE-ONLY
    If a Nearline disk instance exists, the data is transferred from Nearline disk to the destination server. Alternatively, the data is first transferred entirely to Nearline storage on disk from tape, and then transferred to the destination server. If no Nearline service is available, the job will be terminated.
  * NEARLINE&DIRECT
    If Nearline transfer is not available (for example, no actor with Nearline storage is available), a direct transfer is performed instead.

- Request Options
  This is only used if a user name and password are required for the server, otherwise leave this blank.
- Request Comments
  Enter any comments in this field to be sent to the Manager with this action job.
- Pathroot
  Enter the path to the metadata files on the server.
- Object Name Filter
  Enter the Object Name filtering criteria to use for this filter. An asterisk is a wildcard character and the default entry in this field.

- `Object Source Filter`

  Enter the Object Source filtering criteria to use for this filter. An asterisk is a wild-card character and the default entry in this field.

- `Object Collection Filter`

  Enter the object collection filtering criteria to use for this filter. An asterisk is a wildcard character and the default entry in this field.

- `Cascade Delete Links`

  This parameter determines whether to perform a Delete Action on just the original object or on both the original and the metadata object. If set to Y, when the original is deleted, the metadata object will also be deleted. When set to N, when the original is deleted, the metadata object remains untouched.

- `Cascade Restore Links`

  This parameter determines whether to perform a Restore Action on just the original object, or on both the original and the metadata object. If set to Y, when the original is restored, the metadata object will also be restored. When set to N, when the original is restored the metadata object remains untouched.

- `Target Name Template`

  For a metadata archive job, this parameter serves two purposes; it notifies SPM to look for an object with this parameter value on the server, and it notifies SPM to create a metadata archive file on the target using this parameter value. The caret (^) character indicates the system should use the original name.

- `Target Collection Template`

  Enter the Collection for the transcoded clip on the target. The caret (^) character indicates the system should use the original Collection.

6. Click OK to complete the process. The new slot will be displayed in the Slots view under the *Configuration* > *Resources* > *Storage Plans* > *Slots* view.

SPM Delete Object Behavior

Caution: This feature must be managed carefully because objects can be lost.

The `DELETE_OBJECT_ONLY_LAST_INSTANCE` parameter is a configuration parameter that works with the existing `ALLOW_OBJECT_DELETION` parameter.

`ALLOW_OBJECT_DELETION` Parameter

This parameter must be enabled for SPM to delete objects. Delete Object must be enabled or disabled for each Storage Plan separately when configuring them in the DIVA Web App. This configuration method provides additional security and flexibility. Delete Object is enabled only when it is enabled in both the configuration file and the Storage Plan configuration.

Note: If this parameter is disabled, allowing it for a Storage Plan will have no effect.

Delete Object is executed only when the following conditions are satisfied:

- The `ALLOW_OBJECT_DELETION` parameter must be enabled.
- Delete Object must be enabled for the Storage Plan level through the DIVA Web App.
- All of the Storage Slots in the Storage Plan associated with the object must have a Slot End Time.
- Delete Object will be executed by the Storage Slot that has the highest end time in the Storage Plan.

  An object can be associated with only one Storage Plan.

  If multiple Storage Slots have a maximum end time in a Storage Plan, the slot with the maximum Slot ID among the slots with maximum end time will perform the Delete Object.
- Only enabling `ALLOW_OBJECT_DELETION` will not acknowledge a Storage Slot medium Watermark, and the Delete Object is executed at the end of the Storage Slot that has the highest end time in the Storage Plan.

The valid values for this parameter are true or false, and the default is false.

DELETE_OBJECT_ONLY_LAST_INSTANCE Parameter

This parameter adds additional checks and conditions to the SPM Delete Object feature.

Enabling this parameter forces SPM to delete an object at the end of a Storage Slot only if it is the last instance in the entire DIVA system, and that last instance exists on the Storage Slot medium with the highest Slot End Time among all Storage Slots of the Storage Plan. Otherwise, SPM will never execute a Delete Object at the end of the Storage Slot, it will always only execute a Delete Instance at the end of a Storage Slot. The following conditions must be satisfied for this to execute correctly:

- This parameter requires `ALLOW_OBJECT_DELETION` to be enabled. Only enabling `DELETE_OBJECT_ONLY_LAST_INSTANCE` has no effect, and the SPM Delete Object feature will not be enabled.
- The parameter `DELETE_OBJECT_ONLY_LAST_INSTANCE` must be enabled.
- Delete Object must be enabled for the Storage Plan level through the DIVA Web App.
- All of the Storage Slots in the Storage Plan associated with the object must have a Slot End Time.
- Delete Object will be executed by the Storage Slot that has the highest Slot End Time in the Storage Plan.

  An object can be associated with only one Storage Plan.

  If multiple Storage Slots have a maximum end time in a Storage Plan, the slot with maximum slot ID among the slots with maximum end time will perform the Delete Object function.

- At the end of the Storage Slot, if more than two instances exist in the DIVA system or on the medium, only Delete Instance is executed.
- Delete Object is executed only if one instance exists on the entire DIVA system, and that one instance exists on the medium of the Storage Slot that has the highest Slot End Time among all Storage Slots of the storage plan.
- If the medium of the Storage Slot that has the highest Slot End Time is Water-marked, the Delete Object is postponed until the Watermark condition is satisfied.

Storage Plans

Storage Plans contain Slots that can be altered if required. New slots can be added, and existing slots can be edited, deleted, and enabled or disabled. These adjustments to Storage Plans must be performed carefully to ensure the results that are desired. Below is an example of adding a slot to a Storage Plan, and changing an existing slot within the same Storage Plan.

Example:

- DISK1 = Local DIVA Managed Disk

  Slot Type: Storage Slot

  Slot Start Time = 0

  Slot End Time = -1
- NEARLINE = Local Tape Group

  Slot Type: Storage Slot

  Slot Start Time = 0

  Slot End Time = -1
- OFFLINE = Local Tape Group

  Slot Type: Storage Slot

  Slot Start Time = 0

  Slot End Time = -1
- DISK2 = New DIVA Managed Disk

  This is not an initial slot.

  This Storage Slot is added in Step 4 of the following workflow with a Slot Start Time = 0, and Slot End Time = -1.

The company has Storage Plan A, which includes three total instances; one instance on disk and two instances on tape. The following is the workflow for Storage Plan A:

- Object1234 is archived to DISK1 under Storage Plan A.
- Storage Plan A has slots for the following operations, and Object1234 is copied to both tape groups:

  Copy to NEARLINE

  Copy to OFFLINE

- At some point in the future, the company changes its internal policy for content with Storage Plan A to three total instances; two instances on disks (DISK1 and DISK2), and one instance on the NEARLINE tape group.

Adding a Storage Slot

1. Add the new slot change that will copy Object1234 to DISK2.
2. Delete the object instance from the NEARLINE tape group.

   To delete the object instance from the NEARLINE tape group, change the Slot End Time to 0 for the NEARLINE Storage Slot.

   See *Slot Disablement or Deletion*.
3. Add a new Storage Slot on DISK2 with a Slot Start Time = 0 and Slot End Time = -1

Storage Plan A is now updated with the Nearline Slot configuration to delete the instance from the NEARLINE tape group. The Slot Begin Time and Slot End Time are both set equal to 0.

Slot Disablement or Deletion

Telestream recommends you don't delete slots. Deleting a slot doesn't remove the content from the DIVA system. Instead, disable slots.

For a Storage Slot, setting the Slot End Time to 0 forces SPM to delete any contents created by this slot. The Storage Slot can be deleted or disabled after all the contents are removed. To retain the content in a Storage Slot, but no further processing of future actions is required, the slot should be disabled instead of deleting it.

Slot Addition

A slot can be added to an existing Storage Plan. However, caution must be used to achieve the desired results.

When adding a slot to an existing Storage Plan, ensure that the Slot Start Time and Slot End Time are both valid . New slots are always immediately applicable to new objects, but for existing objects ensure that the Slot Start Time and Slot End Time are correct. Otherwise, the action specified in the new slot will not be executed on existing objects.

For valid parameters, see *Configuring Storage Slots*.

Example:

A Storage Plan with numerous objects that have been archived 48 hours before the current time. If a new Storage Slot is added, and is configured with a Slot Start Time = 0 and a Slot End Time = 24, the new slot's action will be added but not applied to existing objects because the slot has already expired for all of the existing objects:

```
((Archive Object Time + Slot End Time (24 hours)) < Current Time)
```

For existing objects to be affected by the new Storage Slot, they must be immediately copied to the Storage Slot medium, but never deleted. To perform this function, set the

Slot Start Time to 0, and the Slot End Time to -1 (infinite). As a result, all instances will be copied to the Storage Slot medium:

```
((Archive Object Time + Slot End Time (infinite)) > Current Time)
```

For non-Storage Slots (for example, Restore Slots), configure the Slot Start Time = 0 and the Slot End Time = 100. In this case, SPM will try restoring objects that have existed for less than the Slot End Time (100 hours); that is, the objects whose Slot End Time has not expired:

```
((Archive Object Time + Slot End Time (100 hours)) > Current Time)
```

If SPM is not able to complete the restore of the object within 100 hours from the archive time, that object is never restored. If the Slot End Time = -1, SPM will try to restore all of the objects.

Watermark-Based Disk Cleaning Management

Disk-array cleaning can be managed by a fixed slot duration, where disk instances or objects are deleted when they expire, or using a Watermark based housekeeping mechanism, where disk instances or objects are only deleted when space must be recovered. This section discusses the details of the second option.

Based on configuration, SPM may attempt to delete instances, or objects. When a Storage Slot's retention for an object is expired, and the object has no more open Storage Slots at the time, SPM will attempt to delete the object if the configuration authorizes it. Otherwise, SPM will only attempt to delete the instance located on the medium referenced in the slot.

The order in which array instances, or objects, or both, of content are removed during the DIVA Array Housekeeping activities can be explicitly specified. SPM includes an array monitoring component to manage array space usage called Watermarking.

Array housekeeping is based on a Watermark system monitored by the Disk Space Monitor component. When the array's Storage Slot retention period for an object expires, the instance, or object, or both, becomes eligible for deletion, but stays on the array until there not enough space available.

Note: Watermarking is only valid for Storage Slots.

High Watermark

DIVA initiates housekeeping activities and expired instances, or objects, or both are deleted according to the defined policies when this threshold is reached. If no instances or objects are eligible for deletion (for example, the retention period has not expired, or a dependency is not met), used array space can exceed the watermark. It is the user's responsibility to ensure that the array filling rate matches the system configuration, and to monitor how array use evolves.

Low Watermark

DIVA stops deleting instances and objects at the Low Watermark. Supported policies and rules for housekeeping include deleting array instances based on Last Access Time or Largest Object Size.

Mixed Mode

When an array is watermarked in Mixed Mode, both instances and objects marked for deletion, and objects whose slots are still open, are considered. Instances and objects in that array are deleted based on whichever of the following comes first:

Note: The Once Only must be set to Y (Yes) in the associated slot's configuration for proper functionality.

- Array reached the High Watermark. Even if the object's slot has not ended, instances and objects will be deleted to bring the array to the Low Watermark level.
- Storage Slot for that object has ended. Even though the High Watermark is not reached, because the slot ended, the instance, or object, or both will be deleted.
- This doesn't mean the array will be cleaned to the level of Low Watermark.

For SPM to delete an object belonging to a Storage Plan, the ALLOW_OBJECT_DELETION parameter must be set to Y in both the SPM Configuration File and DIVA Web App.

External Storage Plans

External Storage Plan functionality allows management of the DIVA Object life cycle using third party external systems. The SPM Database API can be used to develop integration of third party components with DIVA and SPM.

Note: SPM, E (External) should only be used by Telestream personnel.

Validating Configuration

You can validate the accuracy of the SPM configuration with the DIVA Web App.

To validate the SPM configuration, do the following:

1. Open the DIVA Web App.
2. Select a menu on the left of the view.

3. From the menu bar, select Tools, and then Validate SPM configuration. Alternatively you can hold Alt+t, and then press the F3 function key.

   The DIVA Web App will verify the configuration of SPM and return a dialog box with one of two results:

   – DIVA Web App verified the SPM configuration successfully and the dialog box indicates no errors were found.

   – DIVA Web App verified the SPM configuration, found errors, and dialog box displays the errors that were discovered.

   For example:

ERROR: Slot Test must not be of type STORAGE, Please mention correct job type in object Filters Configuration.

Operations

SPM Service Command Line Management

The SPM Service can be managed through the Windows Command-line Interface (Start > Run > *cmd*), or using the Windows Services Panel. Telestream recommends using the command-line interface.

```
DIVACore.SPM.exe {command} [options]
```

The following are the commands:

- `install` (or use `-i`)

  This command installs SPM as a Windows Service.

- `uninstall` (or use `-u`)

  This command removes the SPM Windows Service.

- `debug` (or use `-d`)

  This command starts SPM in console mode for troubleshooting purposes.

- `version` (or use `-v`)

  This command displays the SPM release information, and then exits.

- `help` (or use `-h`)

  This command displays help information, and then exits.

The following are the options:

- `--conf` (or use `--f`)

  Use this option to load the settings from a specific configuration file.

  For example, `DIVACore.SPM.exe install --conf C:\DIVA\conf\test.conf`.

- --log

  Use this option to identify where the log files will be written.

  For example, `DIVACore.SPM.exe install --log C:\DIVA\log\file-name.log`

SPM Information in the DIVA Web App

SPM information for objects can be viewed, and failed actions can be rescheduled using DIVA. This section describes how to use the DIVA Web App to access the information and what information is available through this interface.

Start DIVA, and then follow through this section to discover where to view the SPM information required.

- Which Storage Plan is assigned to a storage medium can be viewed on the *Configuration* > *Resources* > *Cloud Storage* > *Cloud Buckets* view by clicking on a listed bucket. This information is available for disk arrays on the *Configuration* > *Resources* > *Disk Storage* > *Arrays* view by clicking on a listed array.

  If a newly archived object doesn't match any filters it is assigned the default Storage Plan. The default Storage Plan is SP_DEFAULT and has no slots (and therefore no actions) associated with it.

- Information concerning all actions established by SPM can be immediately viewed through the *Content Management* > *SPM Actions* view.

  The ID column on the view is the automatically generated internal `SPM ID`.

  The Task Name column displays the Task Name for the action. The `DEFAULT SPM SCHEDULER TASK` name corresponds to internal SPM generated actions. External actions will display a different Task Name.

  The Req. ID column displays the DIVA Request ID. When an action is in the SCHEDULED state in SPM, the initial Request ID will be zero. When the action begins execution, Manager receives a job and generates the DIVA Request ID.

  The remaining columns on this view are self-explanatory.

  The Request Properties can be viewed by clicking on the action, or by clicking the three dots next to the action and selecting *Job Properties* from the resulting context menu. If no Request Mapping exists for the object, the *Job Properties* window doesn't appear.

Note: If the action is aging, the DIVA Job History may have cycled and the Request ID may have been reused by a newer job. In this case, viewing the Job Properties will display a different job other than the one expected.

SPM Actions View Filter

The SPM Actions panel enables filtering which actions to view. Click the filter icon at the top of the view to filter the view. The default asterisk displayed in the text boxes is a

wildcard and signifies that no filtering restriction is being placed on the parameter where it is used (that is, the object name, Collection, Slot name, and Comments).

To filter by Scheduled Date (and time), select the check box next to enable, and then click the calendar icon to select the Begin and End dates and times to view. If the check box next to Schedule Date Being and Schedule Date End are deselected, this filtering function will be unusable.

Additional filtering is possible by Action Status, or Action Type, or both. Use the pull down list under either one (or both) to display a list of options, and select the desired filters using the check box next to each option.

After changing any of the viewing filters, click Apply Filters to refresh the view.

Assigning a New Storage Plan

Storage Plans can only be changed for an object that is assigned the default Storage Plan (SP_DEFAULT). When a new Storage Plan is assigned to an object, SPM sees that object as being a new object in the system because it has a new Storage Plan assigned.

A list of multiple objects can be selected by selecting the check boxes next to each desired object.

To assign a Storage Plan to the selected objects, do the following:

1. Open DIVA.
2. Navigate to the *Content Management* > *Catalog Browsing* view.
3. Select one or more objects to assign a new Storage Plan to using the check box next to each desired object.

Note: To assign a Storage Plan to a single object use the three dots next to the desired object and select Assign Storage Plan form the resulting menu.

---

4. Click the Assign Storage Plan button on the top of the object list. Alternatively, click the three dots next to any object and select Assign Storage Plan from the resulting menu.
5. A new dialog box is displayed allowing selection of the Storage Plan to be assigned. Select the Storage Plan to assign from the list, and then click Assign to assign the new plan to the selected objects.

Rescheduling Failed Actions

When an action fails to execute properly and doesn't complete correctly, SPM will automatically reschedule the action for re-execution at some time in the future.

To reschedule a failed action to run immediately, do the following:

1. Launch the DIVA Web App.
2. Browse to *Content Management* > *SPM Actions*.

3. Select the action that requires rescheduling (multiple actions can be selected—see the previous section for multi-selection tips).
4. Click Reschedule Action from the resulting context menu. A confirmation dialog box will display.
5. On the confirmation dialog box, select Yes to confirm rescheduling the action to execute immediately, or select No to cancel the process.

Failed Action Completion

The status of SPM Failed Actions can be changed to COMPLETED by clicking the three dots next to the action, and then selecting Mark Action Completed from the resulting menu.

Normally, SPM will retry a completed Copy Action if the Once Only option is set to N, and a user manually (or accidentally) deletes the instance that SPM copied before the Storage Slot expires. Also, SPM will normally retry a completed Delete Action if a user manually (or accidentally) copies an instance to the Storage Slot medium after SPM deleted it.

Actions marked as complete by a user will never be retried by SPM. However, a user completed action can be rescheduled by clicking the three dots next to the action, and then selecting Reschedule Action from the resulting menu.

Note: The Mark Action Completed option is only available using the Administrator profile.

Storage Policy Manager Action Result Codes

Result codes are displayed for every processing and completed action in the *Content Management* > *SPM Actions* panel of the DIVA Web App. The Action Result Code is a short text string reflecting the latest available result of action execution.

The following list describes the possible result codes:

- Prefix: `REQ`
  - Description:

    `DIVA_REQUEST_STATE` code from the DIVA API.
  - `Details:`

    SPM always updates the status of an action after execution based on the status of the corresponding job submitted to the Manager. SPM uses the `GET_REQUEST_INFO` API call to find the status of the job, and updates the result code for the action based on the `DIVA_REQUEST_STATE` code returned by `GET_REQUEST_INFO`.

    For detailed information, see the *DIVA Application Programming Guide*.
  - Sample `RESULT_CODE`:

    `REQ-REQUEST_STATUS` where `REQUEST_STATUS` is one of the following:

  * TRANSFERRING
  * MIGRATING
  * COMPLETED
  * ABORTED
  * CANCELLED
  * UNKNOWN
  * DELETING
  * WAITING FOR RESOURCES
  * ASSIGNING POOL
  * PARTIALLY ABORTED
  * RUNNING
  * PENDING

- Prefix: API
  - Description:

    DIVA_STATUS of the DIVA API.
  - Details:

    SPM uses the API to execute actions and submit jobs to the Manager. The result code of the action is updated with the DIVA_STATUS API return code if SPM failed to submit the job.

    For detailed information, see the *DIVA Application Programming Guide*.
  - Sample RESULT_CODE:

    API-RETURN_CODE where RETURN_CODE is one of the following:

    * DIVA_OK
    * DIVA_ERR_UNKNOWN
    * DIVA_ERR_INTERNAL
    * DIVA_ERR_NO_ARCHIVE_SYSTEM
    * DIVA_ERR_BROKEN_CONNECTION
    * DIVA_ERR_DISCONNECTING
    * DIVA_ERR_ALREADY_CONNECTED
    * DIVA_ERR_WRONG_VERSION
    * DIVA_ERR_INVALID_PARAMETER
    * DIVA_ERR_OBJECT_DOESNT_EXIST
    * DIVA_ERR_SEVERAL_OBJECTS
    * DIVA_ERR_NO_SUCH_REQUEST
    * DIVA_ERR_NOT_CANCELABLE
    * DIVA_ERR_SYSTEM_IDLE
    * DIVA_ERR_WRONG_LIST_SIZE
    * DIVA_ERR_LIST_NOT_INITIALIZED
    * DIVA_ERR_OBJECT_ALREADY_EXISTS
    * DIVA_ERR_GROUP_DOESNT_EXIST
    * DIVA_ERR_SOURCE_OR_DESTINATION_DOESNT_EXIST
    * DIVA_WARN_NO_MORE_OBJECTS
    * DIVA_ERR_NOT_CONNECTED
    * DIVA_ERR_GROUP_ALREADY_EXISTS
    * DIVA_ERR_GROUP_IN_USE

  * DIVA_ERR_OBJECT_OFFLINE
  * DIVA_ERR_TIMEOUT
  * DIVA_ERR_LAST_INSTANCE
  * DIVA_ERR_PATH_DESTINATION
  * DIVA_ERR_INSTANCE_DOESNT_EXIST
  * DIVA_ERR_INSTANCE_OFFLINE
  * DIVA_ERR_INSTANCE_MUST_BE_ON_TAPE
  * DIVA_ERR_NO_INSTANCE_TAPE_EXIST
  * DIVA_ERR_OBJECT_IN_USE
  * DIVA_ERR_can't_ACCEPT_MORE_REQUESTS
  * DIVA_ERR_TAPE_DOESNT_EXIST
  * DIVA_ERR_INVALID_INSTANCE_TYPE
  * DIVA_ERR_ACCESS_DENIED
  * DIVA_ERR_OBJECT_PARTIALLY_DELETED
  * DIVA_ERR_LICENSE_DOES_NOT_SUPPORT_THIS_FEATURE
  * DIVA_ERR_COMPONENT_NOT_FOUND
  * DIVA_ERR_OBJECT_IS_LOCKED
  * DIVA_ERR_OBJECT_BEING_ARCHIVED

- Prefix: SPM
  - Description:

    SPM business logic result code.
  - Details:

    The description for each result code is included with the sample codes in the following list.
  - Sample RESULT_CODE:
    * SPM-LR_CONFIG_PREVENTS_DELETE

      ALLOW_OBJECT_DELETION parameter in the SPM.settings.json file was set to false and prevented SPM from deleting the last instance of an object.
    * SPM-LR_can't_DELETE_LAST_INSTANCE

      SPM refused to delete the last instance of an object.
    * SPM-LR_NO_MORE_INSTANCES_ON_SLOT

      SPM refused to create more instances on a slot than it was permitted.
    * SPM-LR_POSTPONED_AS_WATERMARKED

      SPM postponed a Delete Instance Action because it was for a Watermarked media.
    * SPM-LR_NO_NEED_TO_LINK_OBJECTS

      The Verify stage of the LINK OBJECTS step. If the METADATA_ARCHIVE Slot was configured without Cascade Objects for Delete or Restore (both set to N), there will be no link between the objects, and this error will be produced.
    * SPM-LR_can't_SEND_MORE_REQUESTS

      The Manager Monitor doesn't allow sending more jobs to the Manager.

    * `SPM-LR_ACTION_SLOT_EXPIRED`
      An action will have this code either because the object was deleted from DIVA, but is still under SPM, or the slot expired after loading the SPM action into memory.

- Prefix: `USR`
  - Description:

    User action was executed.
  - Details:

    The user manually marked the action complete.
  - Sample `RESULT_CODE`:

    `USR-ACTION` where ACTION can be `MARKED_COMPLETE`.

SPM Logging Feature

Note: Parameters previously included in the spm.trace have been moved to the SPM.settings.json file. The spm.trace file is no longer used.

The SPM Logging configuration is located in the SPM.settings.json file and assists with controlling the SPM log file size and file switching. The parameter descriptions are in the `SPM.settings.json` file.

AVID Connectivity and Avid for DIVA Applications

This document describes the installation, configuration, and operation of all Avid components.

Topics

- About Avid Connectivity
- Installing Avid Connectivity and AvidForDIVArchive
- AMC Installation, Configuration, and Operations
- AWD Installation, Configuration, and Operations
- Avid Archive Manager Updater Tool

About Avid Connectivity

Avid Connectivity interfaces with DIVA to transfer archival data to and from DIVA in specific formats, specifically Material eXchange Format (MXF), and to enable archiving and retrieving one or more clips. Avid Media Services manages all jobs submitted by the Avid GUI (Interplay Access or Media Composer).

Note: TMC (Transfer Manager Communicator) has been discontinued.

Components for the Archive Manager Communicator (AMC) and Avid Workflows for DIVA (AWD) aren't installed with the main DIVA installation. You must install AMC and AWD components separately. You can install either of the following services:

- AMC 2.1 for AVID Direct ISIS (an Avid Shared Storage System) connectivity, using Archive Manager Engine and DMS Archive API interface.
- AWD Service for AVID Direct ISIS connectivity, using Interplay Engine and Interplay Web Services interface.

AMC and AWD also require additional installations. This chapter describes the installation, configuration, and operation of all Avid components.

All operations for the AM Communicator are performed from Avid Interplay, not from DIVA.

DIVA supports the Avid Web Services API; so you can archive, restore, and partially restore clips and sequences directly from Interplay. DIVA also supports AMC for Interplay 2.2 and later.

DIVA doesn't support certain API operations for complex objects in Avid Connectivity, such as `GetByFilename` and `DeleteByFilename`.

Avid Interplay (Avid PAM, Avid MediaCentral Production) Supported Environments

The following are supported for Avid connectivity and Interplay:

- AMC [2.1]: Interplay 2.2 or later: DIVA Core 8.0 or later
- AWD [1.0]: Interplay 3.6 or later: DIVA Core 8.3 or later

AvidForDIVArchive and DIVA Release Compatibility

`AvidForDIVArchive` is compatible with the following DIVA releases:

- The feature AMC 2.1 for direct ISIS is compatible with DIVA Core releases 8.0 and later.
- The feature AWD 1.0 is compatible with DIVA Core releases 8.3 and later.

AM Communicator (AMC)

The AM Communicator (AMC) enables interaction between the Avid Archive Manager solution and DIVA. AMC receives Push, Pull, and Remove jobs from the Avid Media Services and Avid Archive Engine, and translates them into DIVA REST API calls.

Direct ISIS Connectivity AM Communicator

Using Direct ISIS Connectivity, DIVA reads and writes data directly to and from the ISIS server by the actor using an AVID_DIRECT server type. AVID_DIRECT is built on top of the CIFS protocol, and has similar access mechanisms. Therefore Direct ISIS works only with AVID_DIRECT server types.

The ISIS Client must be installed and configured on each actor that reads to, and writes from, the ISIS server; otherwise, the actor can't access the ISIS server. The configuration must match the Archive Provider (AP) configuration. Also, the AVID_DIRECT server must be configured with proper authentication for the ISIS server.

All Restore jobs from Avid are converted into Partial File Restore jobs. Therefore, AMC never submits full Restore jobs to DIVA, because the actor requires the complete ISIS path for files. If a full restore is required, Avid sends a Partial File Restore job with `[Begin, End]` to accommodate the full path requirement.

Metadata AAF files are archived as part of the object (as in Legacy Mode) if the *Allow Metadata Archive* value is set to *Yes*.

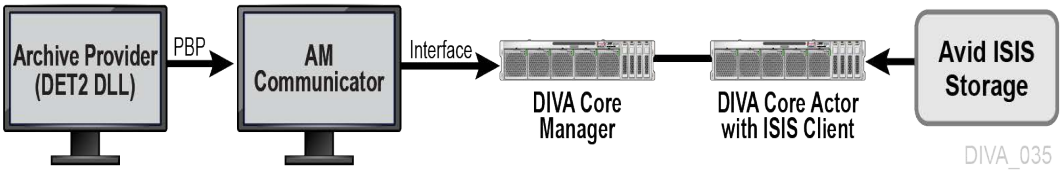
Configure this value in the Archive Manager under
Interplay Administrator > Site Settings > Asset Tracking/Archive Settings.

You can increase the throughput using dedicated network adapter cards. Telestream recommends Intel Pro for 1 Gbps, and Myricom for 10 Gbps, for the ISIS server interface on the actor computers. You can increase overall throughput by increasing the number of actors, limited only by the ISIS server bandwidth. This greatly increases the number of simultaneous jobs that can be submitted by Avid, AMC and fpdiDET2 are *processing* commands. You can configure this value in amc.conf. The default setting is 100.

- Data flows through a single path using Avid Direct Mode compared to the multiple paths used by a legacy configuration so there are no data bottlenecks.
- The actor reads and writes data to the ISIS server through the ISIS Client rather than the legacy's fpdiDET2 (AP - Archive Provider).
- All features of the legacy workflows are retained except the data path.
- One AMC, and one Backup AMC, are enough per site.
- AMC can be installed on any computer, and AMC doesn't need to be on an actor or archive provider computer (for example, a DIVA computer).
- ISIS Clients installed on all actors interfacing with ISIS should have separate network adapter card cards to obtain the best throughput.
- All objects archived using Legacy AMC are compatible with both the new AMC, and fpdiDET2.

The following figure demonstrates the standard workflow for AM Communicator Push job (Archive) processing using Direct ISIS Connectivity.

For detailed information, see *Direct Connect AMC Workflows*



Avid Workflows for DIVA (AWD)

AWD is a Windows executable running as a service. AWD links the Avid environment, and Avid Shared Storage, with DIVA.

The Avid environment is the Interplay Engine, connected via Interplay web services. Avid Shared Storage is either ISIS or Nexis. AWD links the Avid environment and Avid Shared Storage to DIVA via the DIVA Rest API.

Users can work with Avid to archive Avid assets or restore media files to and from DIVA, by dropping assets to specific folders monitored by AWD. Other working scenarios also are possible.

Installing Avid Connectivity and AvidForDIVArchive

To install Avid Connectivity using the AvidForDIVArchive installation wizard, do the following:

1. Double-click the installation executable file.

   The *Choose Components* view is displayed first.
2. Select the check boxes for the desired feature(s).
3. Click *Next*.
4. Enter the full path in the *Destination Folder* field, or click the *Browse* button and browse to the desired destination folder.

   Telestream recommends installing in the default folder.
5. To proceed with the installation, click *Install*.
6. (Optional) To view the details of the installation process, click *Show Details*.

   The wizard displays the final window.
7. To complete the installation, click *Close*.

AMC Installation, Configuration, and Operations

This section describes the installation, updating, and operation of AMC.

Installation Overview

You can configure interaction between DIVA and Avid Archive Manager using the AM Communicator. Separate configuration stages are required for DIVA, AMC, and Avid Archive Manager servers.

AMC can be installed on any computer, provided the computer is accessible through the network from the actor, DIVA, and Avid Archive Manager servers. This option enables installation of AMC on its own computer, separate from any specific server, if desired.

The following definitions apply when the term or acronym is used within this chapter:

Variable	Description
`$DLL_HOME`	Identifies the folder on target Avid server where the DET2.DLL is installed.
`$AMC_HOST`	Identifies the DNS (Domain Name Service) name or IP address of the computer where AM Communicator is installed.
`$DIVA_HOME`	Identifies the folder created on the target computer during the installation.
`$AMC_HOME`	Identifies the AM Communicator folder on the target computer.
`$AMC_INSTALLATION`	Identifies the AM Communicator folder created during AvidForDIVArchive, with the AMC feature installation as follows: `$DIVA_HOME/AMCommunicatorI22` for Interplay 2.2 and later.

Installing AM Communicator

To install the AM Communicator, do the following:

Make sure the AMC feature was selected when `AvidForDIVArchive` was installed.

1. Copy all files from the `$AMC_INSTALLATION/bin` folder to the `$AMC_HOME` folder on the target server computer.
2. The default configuration file is located in the `$DIVA_HOME/conf/amc` folder. Rename the file named `amc.conf.ini` to `amc.conf`.

   If multiple AMC instances are required to run on the target computer, add the corresponding service name to the file name. The service name must match the `SERVICE_NAME` parameter value from the AMC configuration; that is, rename the file to `amc_$SERVICE_NAME.conf`.
3. Edit the AMC configuration file using a plain text editor (for example, Notepad or Notepad++) to modify any required parameter values.

   For detailed parameter descriptions, see *AMC Installation, Configuration, and Operations* .

Command Line Options

You can start AM Communicator either as a Windows system service or as a console application. Telestream recommends running AMC as a system service in production

environments. However, console mode is useful when performing troubleshooting activities.

If two AMC instances must be run on a single computer, the fully qualified configuration file path must be specified in the command-line using the `-conf` (or `-f`) option described in the following list.

The general syntax to start AMC is as follows:

```
AMCService {options_from_the_following_list}
```

For example, the following command starts an AMC instance using parameters from the configuration file named amc_example.conf, and runs in console mode (the option). The `SERVICE_NAME` parameter value must be set for this sample.

```
AMCService -d -conf ../conf/amc/amc_example.conf
```

For all AMC parameters, see *AMC Installation, Configuration, and Operations*.

AM Communicator supports the following command-line parameters:

Parameter	Description
`install or -i`	Installs AMC as a Windows service using the provided command-line options as the default parameters. This option will not start the service, it only installs the service.
`uninstall or -u`	Removes AMC from the Windows services. This option will stop the service automatically if it is running.
`debug or -d`	Starts the AMC service in console mode. The console mode provides additional information as standard output.
`version or -v`	Displays the AMC release level, and then exits.
`help or -h`	Displays use information, and then exits.
`conf or -f`	Specifies a fully qualified configuration file path for AMC to use instead of the default configuration file. You use this option when running multiple instances of AMC on a single computer (as shown in the previous example).

Configuring AMC for Direct ISIS/Nexis Connectivity

Caution: Never change any Category parameters, for example, DIVA_DEFAULT_CATEGORY when the system has been used previously. This will affect all Restore operations.

The default configuration file named `C:\DIVA\Program\conf\amc\amc.conf.ini` is delivered with specific parameters Direct Connect. The file must be configured with the appropriate parameters. If you are upgrading from a previous release, copy amc.conf and then add any new parameters for the release to the new file.

Installing the ISIS Client

The newest ISIS client must be installed on the actor computer that interfaces with the ISIS server. Execute the ISIS client installation file, and install the program with the default settings.

If the ISIS server is on the same subnet as the actor where the ISIS client is installed, the ISIS client will automatically detect the server. Right-click the server name in the top Systems panel, and select Connect from the resulting context menu to connect to the server.

If the ISIS server is on a different subnet than the actor computer where the ISIS client is installed, to add the server to the ISIS client interface, do the following:

1. Open the ISIS client and click the *Add Server* (globe) button in the icon bar.

   DIVA displays the Remote Hosts dialog.
2. Enter the IP address of the remote ISIS server in the field and click + to add the host name to the Host panel.
3. Confirm that the server has connectivity, and has been added, by clicking the Network icon on the icon bar (the icon looks like two computers connected together).

   A window is displayed showing the added server on the network. Confirm that the Enabled check box is selected to use the server.
4. Right-click the server name in the top Systems panel, and select Connect from the resulting context menu to connect to the server.

AM Communicator Server Configuration Parameters

For Direct ISIS, leave the Server Root Path field empty. There is only one Server type, AVID_DIRECT, for all actors. Multiple actors don't require METASOURCE.

All actors with an ISIS/Nexis Client and the AVID_DIRECT server must be in a single Network so that other (Legacy AMC) actors aren't used with this server.

The other parameters are set as normal for the environment and operations.

Use the following parameters when adding a server for use with Direct ISIS AM Communicator.

Parameter	Description
Source Name	This parameter identifies the AMC server name. The name must either match the value of the DIVA_SRCDST parameter from the AMC configuration, or the partition parameter value from the Archive Profile in Avid (depending on configuration).
Source Type	This parameter identifies the source server type and must be set to AVID_DIRECT for Direct ISIS AM Communicator operations.
Connect Options	The Connect Options field must include the login and password for the server ISIS server. An example entry in this field might be `-login amcdirect -pass is-admin`. The `amc.conf` configuration file must be configured with all required parameters. Important parameters for Direct Connect configuration include the following: • `AM_MAX_CONNECTIONS` - the default is 100. • `DIVA_SRCDEST` - this is a type of AVID_DIRECT. • `DIVA_DEFAULT_CATEGORY` Set all other parameters the same as a Legacy configuration.

Note: The `AM_TIMEOUT` parameter is no longer used for status because the status is not monitored by AMC. The `AM_TIMEOUT` parameter continues to monitor jobs until they are canceled or terminated.

See *AM Communicator Configuration Parameters* for the default, delivered `amc.conf` file.

Configuring Avid Archive Manager

The Push, Pull, and Remove jobs are initiated, and processed by, three separate Avid services. The services normally reside on different computers. In a typical configuration,

Archive and Restore services are installed on the Archive Provider server, and the Archive Engine server handles Remove services.

The DET2 DLL must be installed on the servers where the corresponding Avid services reside to enable Archive, Restore, Partial File Restore, and Delete functionality.

The following procedure is identical for all Avid servers (assuming the `AvidForDIVArchive` AMC feature was installed on the target server):

1. Copy the `fpdiDET2.dll`, and all other files, from the `$AMC_INSTALLATION/fpdiDET2` folder to `$DLL_HOME` folder on the target Avid server. Telestream recommends using the default `C:\DET2\fpdiDET2` path.
2. If the selected `$DLL_HOME` folder is different from the recommended default path, the `am_regkey.reg` file's VendorFolder Key Value must be edited so it matches the `$DLL_HOME` value.
3. Merge the `am_regkey.reg` file into the Windows Registry.
4. Change the AMC parameter value in `fpdiDET2.xml` to define the correct AM Communicator network address using the format {IP}:{Port}.
   - IP

     This parameter is the IP address of the AM Communicator Listener. This must be consistent with the value of the AM_IP parameter from the AMC configuration file.
   - Port

     This parameter is the port number for the AM Communicator Listener. This must match the value of the AM_PORT parameter from the AMC configuration file.
5. If required, set the optional `SOCKET_BUFFER_SIZE` parameter in `fpdiDET2.xml` to enable TCP/IP Scaling.

   This parameter value specifies the socket receive and send buffer sizes in kilobytes. The value must match the value of the `AM_SOCKET_BUFFER_SIZE` parameter from the AMC configuration.

   Leave this parameter set to the default value if TCP/IP Scaling is not configured for your system.
6. The Archive and Restore Provider services must both be restarted for these changes to take effect. Telestream also recommends restarting the Avid Archive Engine server.

Both the Archive and Restore profiles must be defined using the Avid Interplay Media Services and Transfer Status utility to submit Archive, Restore, or Partial File Restore jobs to the AM Communicator.

The partition parameter value defines either the destination server tape group or disk array name, or the server Name in DIVA. Creation and use of multiple Archive profiles enables archiving to different tape groups, disk arrays, or both.

Using the partition parameter as the Server Name enables the creation of Storage Plans based on the server in DIVA. You can also use the Storage Plan as media.

Changing or adding Archive profiles doesn't require AMC Service reconfiguration or restarting.

If the partition parameter value is not specified, or a job is submitted with no profile selected, AMC uses the tape group, or disk array, and Server Names from its configuration file.

Note: All other parameters in the Archive and Restore profiles depend on your specific requirements and are left to your discretion.

Enabling Best Effort Restore

Use the following procedure to enable Best Effort Restore:

1. Open the Avid Interplay Administrator application.
2. Connect to the AvidWG database by clicking the name in the navigation tree on the left side of the view.
3. Select the check box next to Use best effort restore option to enable the function. Deselect the check box to disable the function.
4. Click Apply Changes to save the changes.

Configuring Avid Interplay to Store AAF Files

Avid Interplay 1.4 and later can be configured to store AAF (Advanced Authoring Format) Metadata with the media files. This configuration enables duplicate media files in a single archive.

To enable these options, do the following:

1. Open the Avid Interplay Administrator application.
2. Connect to the AvidAM database by clicking the name in the navigation tree on the left side of the view.
3. Click the Asset Tracking/Archive Settings link.
4. Select the value for the options in the Archive and Restore Settings section of the view.

   Enabling the Allow duplicate media file(s) option consumes more storage space in DIVA, but generally decreases the time required for archived object restoration.

   In order to archive meta-data files (.AAF) as part of the object, select the setting for Allow Metadata Archive option parameter to Yes. This requires updating the `fpdi-DET2.xml` file for `NETWORK_STORAGE_PATH`, which allows the Archive Provider to place the .aaf file in a folder where actor can connect and add it to the object.

   Telestream recommends enabling these two parameters if using these objects with Content App workflows to create proxies by transcoding clips/sequences. Moreover, the Allow Metadata Archive option can be mandatory, following the AMC specific option in the configuration file.
5. To save the changes, click *Apply Changes*.

Avid Interplay 1.6 and later can be configured to enable Best Effort Restore. When the Best Effort Restore option is enabled, AMC attempts to restore as many media files as it can, and reports an error only if can't restore any files at all.

Caution: It is recommended to disable Best Effort Restore because the Avid Interplay Media Services, and Transfer Status GUIs, do not show partially restored, and completely restored clips and sequences differently. As a result, it may appear that some content is successfully restored while it actually is not.

Operations

All operations for the AM Communicator are performed from Avid Interplay, not from DIVA.

Archiving

You use the following procedure to perform Archive operations using AM Communicator:

1. Open the Avid Interplay Access application.
2. Select the required clip.
3. Right-click the clip, and select *Archive* from the resulting context menu.

   The *Archive* dialog box will be displayed.
4. Select the user profile from the *User Profile* list.
5. To begin the Archive operation, click *OK*.

You can check the status of the job using the Avid Interplay Media Services window.

Restoring

You use the following procedure to perform Restore operations using AM Communicator:

1. Open the Avid Interplay Access application.
2. Open the Archive folder in the navigation tree on the left side of the view.
3. Select the archived object requiring restoration.
4. Right-click the clip and select Restore from the resulting context menu.

   The Restore dialog box will be displayed.
5. Select the user profile from the User Profile list.
6. To begin the Restore operation, click *OK*.

You can check the status of the job using the Avid Interplay Media Services window.

Deleting

You use the following procedure to perform Delete operations using AM Communicator:

1. Open the Avid Interplay Access application.
2. Open the Archive folder in the navigation tree on the left side of the view.
3. Select the archived object requiring deletion.
4. Right-click the clip and select Delete from the resulting context menu.

   The *Delete* dialog box will be displayed.
5. Select *All*.
6. To begin the Delete operation, click *OK*.

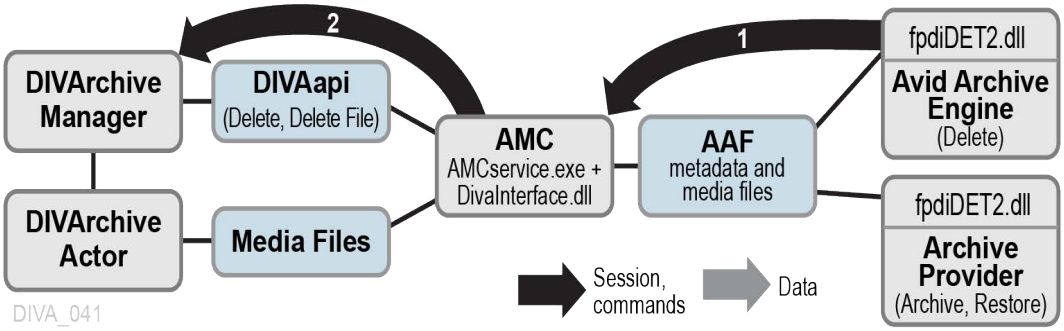
Workflows

This section identifies the various AMC workflows.

Direct Connect AMC Workflows

The following figures are examples of standard Direct ISIS AMC workflows for various jobs (Archive, Restore, and Partial File Restore), followed by a description of the sequence of events for each job.

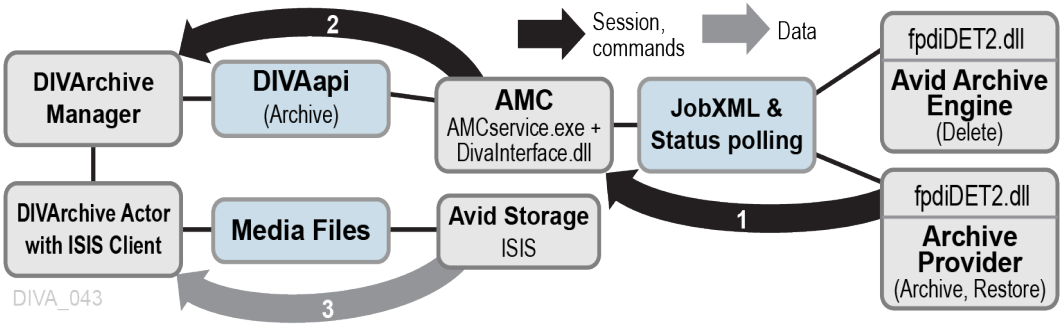
AMC Delete Workflow



When a required Delete operation is identified by AMC, the general Delete workflow runs as follows:

1. The Avid Archive Engine sends a Delete command to the AMC service.
2. The AMC service notifies DIVA that a delete is required, and then DIVA processes the job.

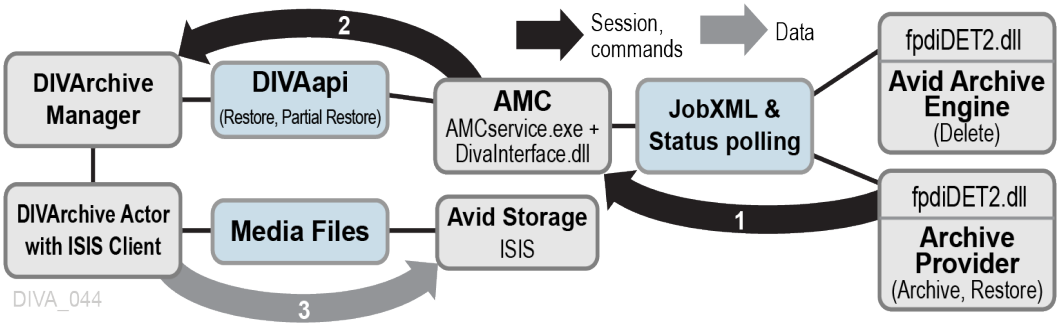
Direct Connect AMC Archive Workflow



The Archive workflow runs as follows:

1. The Media Services Application submits a JobXML to the Archive service.
   a. The Archive Service loads the fpdiDET2.dll and DET2 calls provided.
   b. The fpdiDET2 uses the Avid Archive SDK for this purpose.
   c. The fpdiDET2 connects to AMC, and then submits the Job XML.
2. AMC receives the command, and then receives the Job XML from fpdiDET2.
3. The Job XML is parsed and a list of all files, and their details, is constructed.
4. AMC confirms whether the clip (or sequence of clips) is part of any existing object, determines if deletion is required (based on the configuration), and then sends the Archive job to DIVA.
5. DIVA processes the jobs, and then forwards all necessary information to the actor (using the ISIS/Nexis Client).
6. The actor then archives the objects directly from the ISIS server.

Direct Connect AMC Restore / Partial File Restore Workflow



The Restore workflow is as follows:

1. The Media Services Application submits a JobXML to the Restore service.
   a. The Restore service loads the fpdiDET2.dll and DET2 calls provided.
   b. The fpdiDET2 connects to AMC, and then submits the JobXML.
2. AMC receives the command, and then receives the Job XML from fpdiDET2.
3. The Job XML and fpdiDET2 provide AMC with the file path and MOB ID to construct the DIVA job.

4. AMC searches all categories, identifies a list of objects to Restore, and then sends the Restore job to DIVA.
5. DIVA notifies the actor of the job, and then the actor proceeds to restore the objects directly to the ISIS/Nexis server.

AM Communicator Configuration Parameters

This section identifies the AM Communicator parameters located in the AMC configuration file. Any parameter not specifically mentioned in the following table is used solely for debugging purposes and should not be modified from its default value.

All listed parameters must be defined even if the default values aren't provided in the configuration file.

You can replace all IP addresses mentioned in this section with corresponding DNS names, provided the DNS service functions properly. If the Listener's IP address has the default value 0.0.0.0, then AM Communicator listens to all configured network interfaces on the given computer.

Note: You can find an always up-to-date list of the AM Communicator parameters in the configuration file itself.
The default AMC configuration file delivered with the `AvidForDIVArchive` installer is named `amc.conf.ini`.

Parameter	Description	Default Value
`SERVICE_NAME`	Specifies the name of the AMC service. This option is required to run multiple AMC instances on a single computer. If not specified, the default service name, *DIVA AMCommunicator*, is used.	DIVA AMC
`AM_IP`	Specifies the IP address of the AM Communicator PBP (Play Back Protocol) Listener.	0.0.0.0
`AM_PORT`	Specifies port number of the AM Communicator PBP Listener	6101
`AM_TIMEOUT`	Specifies the time (in seconds) that AM Communicator waits for a connection from the actor.	30
`AM_MAX_CONNECTIONS`	Specifies the maximum simultaneous number of jobs submitted by Avid.	100

Parameter	Description	Default Value
AM_BUFFER_SIZE	Specifies the size of the network I/O Buffer in kilobytes. The valid value range is 8-2048 kilobytes. Telestream recommends keeping the default value for environments with high speed networks.	2048
AM_SOCKET_BUFFER_SIZE	Specifies the socket receive and send buffer sizes in kilobytes. If set to 0, the system default values are used. You can use this parameter to enable TCP/IP scaling if it is supported by the installed hardware and operating system.	0
AM_SOCKET_LINGER	Identifies the maximum time (in seconds) the FTP data socket remains open during a Socket Close call to enable sending of queued data.	1
AM_TREAT_STAGING_AS_ERROR	Sets the behavior of restoring from a Deep Cloud Archive Tier. Instead of waiting for the download to complete (default), this parameter set to 1 indicates to abort the restore job and provides the estimated retrieval duration.	0
AM_IS_AAF_REQUIRED	When set to 1 (default), the archiving submission from Avid must contain the AAF metadata file, and `Allow Metadata Archive` in the Avid Archive settings must be set accordingly.	1
AM_OPTIMIZE_FOR_MIGRATION	Notifies AMC to restore the content in Single File Per Object Mode before searching for objects by file name. You should set this parameter to 1 only when content archived in Single File Per Object Mode is used; that is, most often to speed up content migration to Multiple File Per object format.	0

Parameter	Description	Default Value
AM_READABLE_NAMES	When set to 1, the original clip or sequence name is used as the object name rather than the Metadata MOB ID. This parameter doesn't affect Single File Per Object Mode.	0
AM_OVERWRITE	Identifies whether AMC attempts to delete an existing object, or returns an error if an object with the same name already exists in DIVA. This parameter doesn't affect Single File Per Object Mode.	0
DIVA_MANAGER_ADDRESS	Specifies the IP address of DIVA where AM Communicator sends jobs. You can specify multiple IPs separated by commas. AMC attempts to connect to the second IP if it can't connect to the first. You must configure the same port number on all DIVAs.	localhost
DIVA_MANAGER_PORT	Specifies the DIVA port where AM Communicator sends jobs.	9000
DIVA_REQUEST_RETRY	Specifies the number of reconnect attempts made if a job sent by AM Communicator fails due to a lost connection with DIVA. Telestream recommends leaving this value at the default setting.	1
DIVA_MONITOR_INTERVAL	Specifies the job status polling interval (in seconds).	10
DIVA_SRCDST	Specifies the name of the AMC server configured in DIVA.	
AM_CATEGORY_DELIMITER	Specifies the delimiter separating the base Collection from the appended resolution, and optional index. Changing this parameter after it has been set may prevent previously archived objects from being successfully restored.	~

Parameter	Description	Default Value
AM_CATEGORY_IGNORE_LIST	Specifies a comma separated list of resolution values that do not affect object Collection selection.	Metadata, PCM, WAVE, AIFC, MPEG1Layer2
AM_CATEGORY_MIXED_RESOLUTION	Specifies a special resolution value to construct the Collection for multi-resolution content. Telestream strongly recommends keeping this parameter value empty if AMC is upgraded from an earlier implementation.	
DIVA_DEFAULT_CATEGORY	Specifies the base Collection of objects created by AM Communicator.	
DIVA_ARCHIVE_PRIORITY	Specifies the priority of jobs initiated by AM Communicator.	-1
DIVA_ARCHIVE_QOS	Specifies the value of the Quality Of Service parameter for Archive jobs initiated by AM Communicator.	0
DIVA_RESTORE_QOS	Specifies the value of the Quality Of Service parameter for Restore, and Partial File Restore jobs initiated by AM Communicator.	0
PARTITION_IS_MEDIA	When set to 1, the partition parameter from the Archive Profile specifies the destination server media. When the value is set to 0, the partition parameter value specifies the server name.	1
DIVA_ARCHIVE_MEDIA	Specifies the name of the media where objects are archived. Use this parameter in configurations where the partition parameter value identifies the server, and when the archive is submitted with no profile selected.	

AWD Installation, Configuration, and Operations

This chapter describes installation and configuration of Avid Workflows for DIVA (AWD). AWD interacts with Avid via Avid Interplay Web Services and with DIVA via the DIVA Rest API.

Installing Avid Workflow for DIVA (AWD)

Avid Workflow for DIVA (AWD) installation requires the *AvidForDIVArchive* installer, release 3.0 or later.

To install AWD, do the following:

1. Download the installer from the DIVA releases repository.
2. Launch the installer and select `Avid Workflows for DIVA`. The AMC feature can be unchecked, because it is not used in this specific case.
3. Select the target folder (usually `C:\DIVA`. This folder must exist).
4. After successful installation, the service DIVA AWD Service is automatically installed. The service is not started yet; it needs to be configured first.

Note: The batch script is installed in the `bin` folder, and can manage the service as usual.

Configuration

The configuration file is `AWDService.settings.json` in the `conf/awd_service` folder. It can be created or updated following the structure of the INI file created by the installer in the same folder. After all the options are set, the service can be started.

While running, the logs are generated in the `log/awd_service` folder.

Note: The INI file should be used only as a reference for configuration.

The configuration has one or more Avid Interplay profiles. See the `AWInterplayOptions` section in the configuration file.

For each Avid Interplay profile, the parameters are as follows:

Parameter	Description
`WSAddress, WSPort`	IP address and port of the Interplay WS.
`WSWorkgroup`	Avid workgroup name for the Interplay WS.
`UserName, UserPassword`	User credentials for Interplay access.

Parameter	Description
`AvidWGFolder`	Path to the IE watch folder, monitored by AWD to detect Avid assets to archive or restore.
`WatcherPeriodInSec`	Minimum interval polling (in seconds) of the watch folder. If less than the default value (5 seconds) the default will be considered.
`AvidWGRunningFolder`	Path to the IE folder where the assets are moved from the watch folder, after they satisfy the constraints imposed by the specific workflow (that is, in the case of archive, all the media files are online). The assets are kept in this folder pending the transfer to or from DIVA; after the transfer completes, they will go to either the Error or Completed folders, based on the final status of the transfer.
`AvidWGErrorFolder`	Path to the IE folder where AWD can move the assets for which the transfer to or from DIVA aborts.
`AvidWGCompletedFolder`	Path to the IE folder where AWD move the assets after completion.
`IsToArchive`	Direction flag: true if the detected assets must be archived, false or missing in the restore case.
`IsDelete`	Delete flag: true if the detected assets must be deleted, false or missing otherwise.
`Partition`	Optional, specifies the media group for the archive. If not set or missing, the media group from Archiver is considered (that is, AWArchiverOptions/Archive/Media). If set, the specified media group must exist.
`SplitSequenceArchive`	Optional, specifies the archive behavior. If it is enabled, a sequence asset must be archived into multiple objects in DIVA, one object for each contained clip and one object for the sequence itself. By default, the asset is archived as single object.
`BestEffortRestore`	Optional, specifies the restore behavior. If enabled, it can allow restore jobs to succeed, even if some individual media files could not be successfully restored. If disabled (default), then the restore will fail on missing required files.

Parameter	Description
`Resolution`	(Optional) Specifies the resolution to restore/delete: **All.** All resolutions. If enabled, all asset's archived resolutions must be restored. It takes precedence over any other resolution option. **Current.** If enabled, only the Avid current resolutions (i.e., last check-in) must be restored. It is implied by All option. **Custom.** Array of values, comma-separated, containing the specific resolutions to restore. Only the media files having the resolution in the specified list will be restored. The Custom option is effective only if neither All nor Current is selected. If no choice is set or the entire option is missing, the default restore resolution All for clip assets and Current for sequence assets will be considered. Note the default delete resolution is none.
`AssetDelete`	**Delete only.** Specifies the asset to remove: **Archived.** If enabled, it must delete the archived object (if exists). **Online.** If enabled, the AVID asset itself must be deleted. In a delete profile, the options Archived, Online and Resolution are cumulative.
`ForceRetryOnCritical Error`	Optional, specifies how to recover from a critical error (for example, after stopping the service or crashing/ restarting the machine). If enabled, forces affected jobs to be retried. Note the retrying is forced when the service restarts, and it takes precedence over possible canceling actions happened in the Archiver UI side while the service was not running. If disabled (default), then jobs that were running at the critical time are moved to the Error folder.
`AvidSharedPath`	UNC path to the ISIS/Nexis folder where the restored media files must go, or in the archive case, the metadata file is delivered.

AWD uses the DIVA Rest API for the communication with the Archiver. A specific section in the configuration file refers options for DIVA access and behavior.

See the `AWArchiverOptions` section in the configuration file.

Parameter	Description
`ApiUrl`	Base URL pointing to the DIVA API gateway.
`AUserName`, `AUserPassword`	User credentials for DIVA API access
`MaxJobs`	The maximum number of parallel jobs allowed.
Archive: Options specific to the Archive workflow.	
`MaxRequests`	The maximum allowed number of DIVA parallel requests per archive job.
`Source`	AWD source name in DIVA. Must be of type `AVID_DIRECT`.
`Media`	Destination media group. Can be superseded by the Partition option set at specific profile level.
`Overwrite`	Indicates whether an existing DIVA object is deleted as a result of archiving an object under the same name and collection.

`QOS`	Values for DIVA_ARCHIVE_QOS:
	0 => DIVA_QOS_DEFAULT // Direct and Cache
	1 => DIVA_QOS_CACHE_ONLY // Cache only
	2 => DIVA_QOS_DIRECT_ONLY // Direct only
	3 => DIVA_QOS_CACHE_AND_DIRECT // Cache and direct (if cache not available)
	4 => DIVA_QOS_DIRECT_AND_CACHE // Direct and Cache (if direct not available)

`Priority`	Priority of submitted archive requests.
Restore: Options specific to the Restore workflow.	
`MaxRequests`	The maximum allowed number of parallel jobs per restore job.
`Destination`	AWD Destination Name in DIVA. Must be of type `AVID_DIRECT`.

Parameter	Description

QOS	Values for DIVA_RESTORE_QOS:
	0 => DIVA_QOS_DEFAULT // Direct and Cache
	3 => DIVA_QOS_CACHE_AND_DIRECT // Cache and direct (if cache not available)
	4 => DIVA_QOS_DIRECT_AND_CACHE // Direct and cache (if direct not available)

Priority	Priority of submitted restore jobs.
Delete: Options specific to the Delete workflow.	
Priority	Priority of submitted delete jobs.

Operations

The following subsections describe typical operations.

The Interplay Profiles

An Interplay profile, as used in AWD, allows setting up templates to use when performing an operation on the Interplay Avid assets.

The AWD determines the operation (or workflow) by the profile type. The profile types, Archive, Restore, and Delete, are defined by the direction flag. The Avid assets subject to the workflow are the ones retrieved from the specific Interplay watch folder.

You can use the Avid UI (for example, Interplay Access) for delivering assets to the watch folder. By example, a drag & drop can easily be used in the Interplay Access. You can also use other methods for delivering assets; for example, checking-in assets from an external application.

The watch folder, as source of assets for a specific workflow, is a pillar element in an Interplay profile. For this reason, a concrete watch folder must be defined in one and only one running profile, and that profile must be unique over all AWD running instances, including the scenario where AWD is installed on multiple machines.

This restriction doesn't apply to the job folders.

The Job Folders

The AWD retrieves assets from the watch folder, triggers a workflow job for the assets satisfying the profile criteria, and removes the assets from the watch folder. The AWD ignores assets that don't meet the criteria.

In the case of a sequence, its referred clips are also submitted to the AW job. This leads to the AW job being represented by a folder (auto-created) containing all the assets involved. During the job execution, the job folders are kept as sub-folders of the

profile's *running folder*. The live status of each job is presented as an annotation in the Comments field of the corresponding job folder.

When the job execution finishes, the job folder is removed from the running folder. In the case of a canceled or aborted job, the job folder is moved in its current state to the *error folder*; the user can then view the error reason in the job folder's Comments field (and perhaps take actions to eliminate the error cause and retry the job). If the job succeeded, the assets involved are moved to the *completed folder* in a flat form; the user can take the post-completion actions (for example, removing the media and/or metadata after a successful archive).

Avid Archive Manager Updater Tool

The AM Updater is used for migrating the AAF files into the Archive Manager (AM) Database. The AM Database is needed for initiating restore jobs from Avid using AMC.

The AMUpdater application is a migration utility designed to import AAF metadata files into the Avid Interplay | Production Archive Engine database. The primary scope is to import the AAF files generated by Flashnet in a scenario without Archive Engine into the database of a new fresh installed Archive Engine.

Notes: In cases where the AAF has not been provided with a Flashnet IPWS archive, restore using Interplay is not possible. A destination located on the ISIS/NEXIS available to the media indexer must be added to the DIVA application. Manual restore jobs of these archives/objects can then be restored to the folder. The Media Indexer picks up the media as if restored via Avid.
Partial File Restore support for assets archived using Flashnet IPWS will not be available due to incomplete archive information presented in the AAF files at the time of archiving (a limitation of IPWS). Partial File Restore of subsequently archived AMC-DIVA assets will be available.

System Requirements

The following are the minimum requirements to run the AM Updater Tool:

- Windows (64-bit)
- 64-bit desktop or server
- .NET framework release 4.5 or later

Installing and Configuring the AM Updater Tool

To install and configure the AM Updater Tool, do the following:

1. Copy the unzipped package to a folder on the local disk.

2. Configure the tool by updating the xml configuration file AMUpdater.exe.config with the actual values in the appSettings section.

   In addition to Avid Interplay | Production Web Services settings (address, port, workgroup, user name and password), configure the following specific application settings:

Setting	Description
`AMUpdater.EnableAvidAMUpdate`	This is a Boolean switch to choose between the effective import (true) and the simulated import (false). In the second case the AAF files are processed without updating the AE database; this option can be useful for detection of invalid AAF files prior import.
`AMUpdater.AvidAMFolder`	The destination folder in Avid AM as the target for the imported assets. In the case of multiple target folders it is the base name for all the target folders. The imported assets are evenly distributed between the target folders.
`AMUpdater.AvidAMFolder.Count`	The number of destination folders in Avid AM. The default is one (a single target folder with the given name). When the option is for multiple target folders the name of the folders are composed by adding a numeric suffix to the base name. If multiple target folders are created (the automatic creation is off), the manual creation of target folders must follow the same pattern for name.
`AMUpdater.AvidAMFolder.CreateIfNotExists`	This is a Boolean switch requesting the automatic creation of the target folders in case they do not exist. If the switch is off (false) then each target folder must exist in Interplay and must be manually created before running the application; otherwise the import will fail. If the switch is on (true) then each target folder will be created by the application unless it already exists.

Setting	Description
`AMUpdater.AAF.SkipForAMA`	This is a Boolean switch requesting to ignore the AMA clip assets. This option should be used for analysis purpose only; it is not recommended in a production migration as it can lead to ignoring valid clips.
`AMUpdater.AAF.SkipForSequence`	This is a Boolean switch requesting to ignore the sequence assets. It can be used in a multiple phase import scenario. Note when this option is active it implies and overrides `AMUpdater.AAF.File.SkipForSequence`.
`AMUpdater.AAF.SkipNotSequence`	This is a Boolean switch requesting to ignore all but sequence assets. It can be used in a multiple phase import scenario. Note when this option is active it implies and overrides `AMUpdater.AAF.SkipForAMA`.
`AMUpdater.AAF.File.SkipForSequence`	This is a Boolean switch requesting to ignore the explicit set of the media files belonging to a sequence. It can be used to minimize the overall import duration. Most parts of these files are assumed to be imported using the referred master clips. When this option is active the computed clips for the rendered effects aren't imported; the effects must be re-rendered again after restore. Note that this option can be used in a multiple phase import scenario.
`AMUpdater.AAF.File.SkipForWellKnownInvalidPath`	This is a Boolean switch (that applies to any kind of asset) requesting to ignore the specific media files having an invalid file path in AAF. These kinds of files can't be set into Avid AM and trying to do so will always generate errors for the files in question.

3. Configure the logging by updating the `system.diagnostics` section of the configuration file:

   The application generates logs. These logs trace the steps of the AMUpdater execution. The file path of the log can be set by updating the initializeData value. The path must exist and must be accessible; otherwise the log is off.

   The log granularity respects the switch setting. Full logging details can be obtained by changing the appTraceSwitch switch value to 4 (Verbose) while the value of 0 (Off) disables the log.

Note: After the migration process is complete, Telestream **strongly recommends** that you uninstall the application by removing it from the computer. The process of importing AAF files should not be used on a regular basis due to the risk of affecting the database integrity.

Usage

After updating the configuration file with the Interplay | Production and application specific settings, run *AMUpdater AAF-Directory* at the command prompt where AAF-Directory is a folder containing (in its sub-folders) the hierarchy the AAF files to be imported.

Frequently Asked Questions

In general, refer to the documentation for the specific component for Frequently Asked Questions about that particular component. Contact Technical Support for any additional questions not covered here.

See *MPEG Disclaimers*.

Topics

- DIVA Operations
- Export / Import Operations
- Analytics

DIVA Operations

Where is DIVA Command, the Configuration Utility, and the Control GUI (Control Panel)?

These three programs have been deprecated and replaced with the DIVA Web App, a Web-based graphical user interface.

What is the sysadmin log in and password?

Contact Telestream. See *MPEG Disclaimers*.

Will an S3 interface be implemented directly to an S3 Object and S3 Glacier, or will there be something between DIVA and storage?

S3 integration is similar to OCI integration, as it is direct to object storage. There is nothing in between the actor and S3 buckets. Telestream provides support for all storage classes, including Standard, Standard-IA, One Zone IA, Intelligent Tiering, Glacier and Deep Archive.

For Glacier, there are 3 tiers of retrievals: Expedited (1-5 min), Standard (3-5 hours) or Bulk (days). It appears that everything defaults to Standard, but a restore can be expedited by changing the Tier parameter of the job. Is this something that will be supported by DIVA, or will everything be standard?

DIVA supports the Expedited tier and the default tier (that is, Standard).

How will PFR work in an AWS environment assuming the customer is using almost exclusively Glacier?

Like any job from Glacier or Deep Archive, DIVA stages the content to Standard prior to transferring from AWS.

How does a user replicate between two regions and have AWS to do it rather than DIVA. For example, to archive content to N. Virginia and then use AWS to move it to a second region, for example, Ohio. Assuming the buckets are configured correctly in both locations, after it is moved, will DIVA be able to discover the content that is now in Ohio?

On archive, DIVA creates a data and metadata bucket and stores AXF files corresponding to the customer's content in those buckets. The newly created disk instance points to the newly created buckets. If the user were to replicate the AXF files from one region to another through AWS, the replicated content would be placed in a uniquely named bucket in the other region. DIVA would not know the name of the new bucket, and therefore would not be able to reference it. However, when support was added for S3 in the Disk Discovery service, it is now possible to scan all buckets in an account and add the content as a new instance.

For OCI, Standard 2.4 VM shapes (4 OCPU, 60GB RAM, 1 TB block storage) were used for actors and DIVAs. Will something equivalent will work on AWS? It appears that maybe an 'i3en.xlarge' would be close with 4 vCPU, 32 GB RAM, 1x2.5TB SSD.

This configuration works fine.

What are the Repack and Verify Tape Job Limitations with Checksum Workflows?

- The repack job will fail if the tape contains any corrupted objects or the object fails checksum verification. The conflict must be manually resolved before performing the repack.
- Repack of WORM media is not automatic. Manual repack is available for WORM media, but the space is not recoverable after repack is complete.
- A checksum is not generated for any spanned objects during Repack or Verify Tape jobs. The actor will identify any spanned files and DIVA will not attempt to verify them. A warning event will be displayed stating that a checksum was not generated or verified for the spanned content. In this case the repack operation will not be terminated, but the object instance will be marked as Not Verified.
- Additional checksum verification is done at the Oracle Storage Cloud level.

  For information, see the Oracle Storage Cloud documentation.

Export / Import Operations

This section includes frequently asked questions about Export/Import Operations. For more information, see *Export and Import Functions*.

What is the export XML and FFM file compatibility?

The exported XML and FFM files, when generated, can be imported into the release of DIVA that they were exported from, and later releases of DIVA. DIVA enables more than one set of tapes (spanned or not) to be exported to and imported from a single file.

What is the Media Type ID?

The Media Type ID is a proprietary DIVA identifier that represents the type of tape media being exported. DIVA exports a `mediaTypeId` field, which corresponds to the Type column in the diva Web App on the *Resources Management* > *Media* view. A Synchronize DB call may be required to update the mediaTypeId, and (or) update the hardware to be compatible with a newly imported tape. Ensure that the block size and total size of the *mediaType* in the source DIVA system matches the *mediaType* definition in the destination. This becomes especially important if the tape is ever repacked.

What are the unsupported DIVA attributes?

- The `markedAsDeleted` is an internal attribute and is not exported or imported through the Export/Import Utility.
- The state of checksum verification (verified, partially verified) is not exported.
- Linked objects and Storage Link information is not exported.
- Information regarding the job that created each object is not exported - newly imported objects aren't associated with a job.

Analytics

How is the Analytics App installed in a new DIVA installation?

The Analytics App is automatically installed with DIVA; no additional installation is required.

Can you choose not to install the Analytics App?

You can choose not to install and run it by not installing and running the DIVA Core Analytics Windows Service. When the service is not started, statistics aren't computed.

Can the Analytics App be disabled?

Yes, you can disable the Analytics App functionality by stopping the Analytics Windows service from running.

How is Engineering Mode accessed?

Telestream has deprecated Engineering Mode, a feature in the legacy Java Control GUI.

DIVA Glossary

Action

A predetermined reaction of a metric surpassing a threshold value by one of the variables from its internal state. This term is used in the Storage Policy Manager associated with a slot and executed during the slot's open period.

Actor

Actors make all types of data transfers, such as Archive, Restore, Copy, and Repack.

See *Actors and Proxy Actors* in the *DIVA Installation and Administration Guide* and *Actors Tab*.

Analytics Service

The Analytics Service monitors the queue, parses the messages and stores them in an Elasticsearch index.

See *Analytics Service Module (Optional)* in the *DIVA Installation and Administration Guide* and Analytics Charts Tab.

APIs (Various)

See *Supported API Releases* in the *DIVA Installation and Administration Guide*.

API Translation Service for Kumulate (Rosetta)

See *API Translation Tab*.

Archive Related Operations Initiator

An entity submitting jobs to DIVA (typically, an automation process).

Archive System Database

This is the DIVA database.

Archive Submissions

An Archive operation is defined as the transfer of files to DIVA. The files are then stored as an object. Archive jobs are issued by navigating to *Content Management* > *Archive Submission* in the DIVA Web App. The form submits an Object Archive job to the Manager for processing.

You can also issue archive jobs using the DIVA Rest API.

For further details, see the DIVA REST API documentation.

The following fields are included in the Archive Submission view:

- Object Name

  The name of the object to be archived.

- Object Collection

  The Collection of the object to be archived.

- Media

  This field designates either a group of tapes, an array of disks, or cloud storage, declared in the configuration where the instance must be created. When this parameter is a null string, DIVA uses the default tape or group of tapes named DEFAULT.

- Storage Plan

  This field defines the Storage Plan to use for this Virtual Object. If no Storage Plan is assigned the default Storage Plan is used.

- Source Server

  The name of the source; for example, a video server or browsing server. This name must be known to the DIVA configuration.

  Define Source Servers in the DIVA Web App in *Configuration* > *Resources* > *System Settings* > *Unmanaged Storage*.

- Files Path Root

  The root folder for the files.

  See the examples in the following section.

- Quality of Service

  One of the following codes :

  – DIVA_QOS_DEFAULT

    Archiving is performed according to the default Quality of Service (direct and cache for archive operations).

  – DIVA_QOS_CACHE_ONLY

    Use cache archive only.

  – DIVA_QOS_DIRECT_ONLY

    Use direct archive only—no disk instance is created.

  – DIVA_QOS_CACHE_AND_DIRECT

    Use cache archive if available, or direct archive if cache archive is not available.

  – DIVA_QOS_DIRECT_AND_CACHE

  For detailed descriptions, see Quality of Service.

  Use direct archive if available, or cache archive if direct archive is not available.

  You can trigger additional and optional behavior for the archive submission via the following options. To request those services, use a logical OR between the previously documented Quality of Service parameter and the following constant:

  – DIVA_ARCHIVE_SERVICE_DELETE_ON_SOURCE

  Delete source files when the tape migration is done. Available for local sources, disk sources, and standard FTP sources. This feature is not available for complex Virtual Objects.

- Options

  Additional options for performing the transfer of data from the source server to DIVA. These options supersede any options specified in the DIVA configuration database. The possible values for Options follow:

  – No Entry

    No entry in this field specifies no options.

  – -r

    Using -r specifies that every name in filenamesList that refers to a folder must be scanned recursively. This also applies when FilesPathRoot is specified and an asterisk designates the files to be archived. This option can be used when archiving from a local source or from a standard FTP server.

  – -login

    A user name and password is required to log in to some sources. This option obsoletes the -gateway option from earlier versions.

  – -pass

    The password used with -login.

- Comments

  Optional information describing the Virtual Object. This field is optional and can be left empty.

- Additional Services

  To delete the original file after it has been archived, select the *Delete On Source* check box.

  Delete on Source is not supported for Broadcast Servers.

- Add New Component

  Enter any additional components to be included in the Archive job. Components are typically source clips, media files or any desired file to be archived into DIVA.

- Priority

  The priority level selection for this job is in the form of a slider control, or text box at the bottom of the form view. The level can be in the range 0 to 100, or the

value Default. If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

The value 0 is the lowest priority and 100 the highest priority. Move the slide control to increase or decrease the job priority, or enter the desired priority number in the Priority Value text box.

There are six predefined values as follows:

- Min
- Low
- Normal
- High
- Max
- Default

  If the Default Priority check box is selected, the slide control becomes inactive and the priority defined in the DIVA configuration is used.

Arrays, Disks, and Cache

DIVA uses HDD (Hard Disk Drive) technologies for both the storage of objects and for transient storage during data transfers (disk cache).

Any disk that DIVA uses is assigned to an array. An array is a logical association of one or more disks for the storage of objects. Disks that are configured as cache disks are also assigned to an array, typically named CACHE.

A disk name is associated with a mounting point. Archive jobs can be submitted with an array as the target. DIVA is responsible for choosing the disk location to write the data when several disks belong to the same array.

AVID Connectivity

Avid Connectivity with DIVA transfers archival data to and from DIVA in specific video formats and enables archiving and retrieval of single clips, or a sequence of clips. Avid Connectivity is no longer packaged with DIVA and is a separate installation process. Additional installation is required for certain plugins for both AMC and TMC.

AXF (Archive eXchange Format)

The Archive eXchange Format 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.

AXF Disk and Tape Storage Formats

AXF (Archive eXchange Format) is an open format that supports interoperability among disparate content storage systems and ensures the content's long-term availability regardless of how storage or file system technology evolves.

Backup Service (BKS)

The DIVA Backup Service (BKS) is responsible for backing up the DIVA Postgres database, the MongoDB Metadata database, and Elasticsearch.

See *Backup Service* in the *DIVA Installation and Administration Guide*.

CA (Certificate Authority)

A CA (Certificate Authority) is an issuer who receives the CSR and returns the SSL certificate with its digital signature.

CAP (Cartridge Access Port)

The Cartridge Access Port is used for inserting and removing tape cartridges to and from a Robotic Tape Library without interrupting library operations.

CAP ID

A unique identifier for the location of a CAP.

Checksum Support and Checksum Types

The purpose of the Checksum Support and Content Verification program is to introduce additional levels of verification into the DIVA system. This feature introduces checksum generation and verification for each file managed by DIVA. Supported checksum algorithms in DIVA include MD5, SH1, MDC2, and RIPEMD160.

A mathematical value computed from a group of data being transmitted, and transferred with the data. The receiving device compares the checksum with its own computation, and if it differs from the received checksum, it requests the transmitting device to resend the data or generates an error. Each checksum has a specific algorithm, each of which has its own level of verification.

For instances of objects that are stored in the cloud, additional checksum is done at the Cloud Storage level.

CIFS

The CIFS (Common Internet File System) is a Microsoft file-system sharing protocol.

Collection

Part of the access key to an object. Collections (referred to as Categories in earlier versions through DIVA 8.x) are an approach to grouping an object with other similar objects having particular shared characteristics. It must not be confused with tape groups, mediums, or arrays, which are storage concepts.

Complex Object

A Complex Object stores more information about the files and folders in an archive, such as subtotals for each directory.

Note: Certain API operations used in Avid Connectivity (such as `GetByFilename` and `DeleteByFilename`) aren't supported for complex objects.

Note: DIVA doesn't support all operations for complex objects: The Delete on Source feature is disabled and the checksum features Verify on Archive and Verify on Restore are also disabled. DIVA Connect doesn't support replication of complex objects.

Complex Objects and FTP

When archiving Complex objects with the FTP protocol and using FileZilla with default settings, the transfer typically fails when archiving any object with more than approximately 3900 files. There are two reasons for this possible failure:

The Actor connection times out before the size of the object can be computed.

A job stops in the middle of the transfer because the FTP server (for example a FileZilla server) is consuming all of the available sockets.

Complex versus Non-Complex Objects

A Complex Object differs from a non-complex object in some key ways. For example, the file and folder metadata information of a Complex Object is stored in the Metadata Database (MDS) not in the Postgres database. The file contains the file names, folder names, checksums, and files sizes. The directory that contains these files is the Metadata Database Root Directory. A Complex Object must be stored in AXF format either on tape or on disk.

Component

A file that is part of an object.

Content Job

System operations that handle archived content — Archive, Copy, Delete, Restore.

Critical Section

A piece of code that accesses a shared resource (data structure or device) that must not be concurrently accessed by more than one execution thread.

CSR (Certificate Signing Request)

A CSR is an encoded file that is given to a CA when requesting an SSL certificate. It contains information that will be included in the certificate including the holder's name, serial number, expiration date and the public key. The CA returns the signed SSL certificate with its signature.

Data Expedition MTP/IP Protocol

A WAN acceleration functionality used for high performance file transfer.

DataBase Agent Service

See DataBase Agent Service in the *DIVA Installation and Administration Guide*.

DET (Dynamically Extensible Transfers)

Dynamically Extensible Transfers are an Avid protocol.

DIVA C++ API (deprecated in favor of DIVA REST API)

Telestream has deprecated the DIVA C++ API and the DIVA Java API.

See *Supported API Releases* in the *DIVA Installation and Administration Guide*.

DIVA Connect

DIVA Connect provides a unified view of archived digital media files across multiple, distributed DIVA systems and the cloud.

For information, see the *DIVA Connect Installation, Administration, and Operations Guide*.

DIVA Connect Complex Object WAN Transfers

DIVA has (optional) WAN acceleration functionality built in that allows it to take full advantage of long distance, high latency, network paths (such as private site to site links or the public Internet), and can perform transfers of complex objects efficiently using the Data Expedition MTP/IP protocol.

DIVA Connect Home Directory

The file system directory where DIVA Connect is installed. DIVA Connect can be installed in its own directory (recommended), or within the DIVA home directory.

DIVA Connect Object (Asset)

An archived object describes digital content in the archive system. A DIVA Connect object is a set of archive objects, archived at one or more DIVA sites, sharing the same name (Object Name and Collection pair). Each object on a particular site can have one or more copies (object instances).

DIVA Connect Site

A sitename uniquely identifies a DIVA Connect site, which is defined as exactly one DIVA installation and one or more DIVA Connect services.

DIVA Database

For DIVA versions up to 8.x this is the Oracle database used by DIVA. Starting with 9.x it is the Postgres database used by DIVA.

DIVA Java API (deprecated in favor of DIVA REST API)

Telestream recommends using the REST API rather than the previous existing APIs (that is, DIVA Enterprise Connect, DIVAS, Java and C++).

See DIVA REST API in the DIVA Installation and Administration Guide.

DIVA REST API

Telestream recommends using the REST API rather than the previous existing APIs (that is, DIVA Enterprise Connect, DIVAS, Java and C++).

See DIVA REST API in the DIVA Installation and Administration Guide.

DIVA Web App

Used for configuring, monitoring and managing the DIVA system. The DIVA Web App is a software GUI utility that connects to DIVA

See *DIVA Web App* in the *DIVA Installation and Administration Guide*.

DNS (Domain Name Service)

A system for naming computers and network services that is organized into a hierarchy of domains. DNS services resolve IP addresses to host names for proper network routing.

DPX (Digital Moving-Picture Exchange)

The DPX format 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.

DSM (Disk Space Monitor)

A module in Storage Policy Manager that assists in array cleanup when the object level reaches the High Watermark.

Drop Folder Monitor (DFM) (optional)

Drop folder monitoring allows users and third party applications to deliver content to be archived by copying related files to a folder, an FTP server, or a CIFS share.

See *Drop Folder Monitor* in the *DIVA Installation and Administration Guide*.

Drop Folder Monitor Incomplete File

Drop Folder Monitor Incomplete files are files that were put into the Watch Folder without the metadata file (for File Set folders), and files that can't be archived after a specific number of attempts.

Drop Folder Monitor (DFM, WFM in Earlier Versions)

The Drop Folder Monitor monitors pre-configured Watch Folders on the system. When new files are detected, one or more operations are performed on the files depending on the folder configuration.

Drop Folder Monitor provides automatic monitoring of newly created files in up to 20 local or FTP folders (or combinations of the two). One (or multiple) files in FTP folders per object are supported. When a new file (or FTP folder) is identified, DFM issues an archive job automatically to DIVA to archive the new file or folders. After the files are successfully archived, they are automatically deleted from the source server.

Event

One operation (such as a job) usually requires multiple events to complete an operation. An event provides all applicable information relating to the single task (for example, names, IDs, parameters, numbers, and so on).

You can view Events in the DIVA Web App in the *Troubleshooting* > *Logged Events* section.

Externalization

An object instance is ejected (externalized) when one of the tapes containing the object's instance elements is ejected. An object is externalized when all of its instances are ejected. An object is considered inserted when at least one instance of the object is inserted.

Filter

Search criteria used in different places of the DIVA Web App to narrow down search results.

FOIB (File Object Information Base)

FOIB is persistent storage used by Drop Folder Monitor to track its processing state.

FQDN (Fully Qualified Domain Name)

The complete domain name for a specific computer, or host, on the Internet. The FQDN consists of two parts: the host name and the domain name. For example, telestream.net.

Initiator

Initiator also refers to the command line available for the BKS module.

See Archive Related Operations Initiator.

Intersite Copy

An operation that copies content from one DIVA site to another, archiving the content to the target site.

Intersite Transfer

An operation that moves content from one site to another, but doesn't archive the content to the target site.

Java Keystore

The JAVA Keystore is a password protected encrypted file repository containing the Key pairs, SSL certificates, and CA certificates.

Jobs

A job is an operation running in DIVA which progresses through steps (migration, transfer, and so on) and ends as either Completed, Aborted, or Canceled.

For details on the various job types available in DIVA, see the *DIVA Administration, Configuration, and Operations Guide*.

The different jobs available under the DIVA Web App's Action menus are as follows.

- Archive
- Associative Copy
- Assign Storage Plan
- Automatic Repack
- Cancel
- Change Priority
- Copy
- Copy As
- Copy Tape
- Delete
- Delete Instance
- Eject Tape
- Export Tape
- Import Tape
- Insert Tape
- Migrate Content
- Multiple Restore (or N-Restore)
- Partial Restore

- Restore
- Repack Tape
- Stage
- Verify Tape

The most common job types are for transferring content to the archive (referred to as an Archive job), or transferring content out of the archive (referred to as a Restore or Partial File Restore job).

In DIVA, a job is a discrete operation on an object. You can issue jobs directly in DIVA, or via an archive initiator.

You use other job types for managing the objects within the archive after they have been created. Examples of other job types are Copy, Delete, and Repack Tape jobs.

Each job is automatically given a unique identifier by DIVA (called the Job ID), which can be used later to retrieve event logs or other properties of each job. DIVA stores the records of up to 500,000 jobs in its database. Then there is a rollover of the job id.

Because multiple jobs may be received simultaneously by DIVA, they are all placed into a queue and are executed on a first come, first served basis. You can prioritize the execution order of jobs using the Job Priority parameter.

The *Content Management* > *Jobs* view in the DIVA Web App displays the queue of jobs that are currently being processed by DIVA. If you use a specific layout for identifying the location of the feature in the DIVA Web App, it should also be used here.

The *Content Management* > *Job History* view displays previously-executed jobs, and the status of those jobs. If you use a specific layout for identifying the location of the feature in the DIVA Web App, it should also be used here.

The DIVA available Job Options are as follows:

- Archive Jobs

  -delete_on_source

- Restore Jobs

  -do_not_overwrite

  -do_not_check_existence

  -delete_and_write

Job Options take precedence over the normal Additional Service specification. Also, the normal Additional Service specification takes precedence over the server Connect Options.

The Additional Services available for a Restore job can also be specified in the server Connect options. If specified, the server uses the Additional Service setting as a default. You can over ride this by specifying the Additional Service at a job level in the normal way, or as a new Job Option. Because these connect options are specific to a Restore job, the options are ignored for any other type of jobs using the server.

Job Types

When connected to DIVA, access the *Content Management* > *Catalog Browsing* view in the DIVA Web App to execute jobs to be issued to DIVA. Click the three dots next to the

desired object to display the context menu with the various job types for that object. A third party initiator application (for example, an Automation System) can be used instead of, or in addition to, the DIVA Web App interface. The available options in the context menu redetermined by the logged in user's assigned role in the user profile.

The different options available from an object's context menu are as follows:

- Search in Job History

Displays the job history for the selected object.

- Assign Storage Plan

Enable assigning the object to a SPM storage plan.

- Restore Object

Displays a form to configure copying a file from DIVA to a single target.

- Partial Restore

Displays a form to select which file, or files, specifically to copy from DIVA to a target server.

- Multiple Restore

Displays a form to configure restoring an object from DIVA to more than one target simultaneously.

- Delete Object

Displays a form to configure deleting all instances, or a selected instance, of an object.

- Copy

Displays a form to configure copying an existing object to another tape group.

- Copy As

Displays a form to configure copying an existing object to another name, tape group, or Collection.

- Stage

Displays a form to configure staging content in the cloud so that it is readily available when Copy and Restore jobs are submitted for the same content.

Journal

A self-maintained, automated, and configurable storage for events.

Key Pair

A Key Pair consists of two uniquely related cryptographic keys; a Public Key and a Private Key (basically long random numbers).

The Public Key is what its name suggests - Public. It is made available to everyone through a publicly accessible repository or directory.

The Private Key must remain confidential to its respective owner. Because the key pair is mathematically related, whatever is encrypted with a Public Key can only be decrypted with its corresponding Private Key, and vice versa.

Legacy Format

Proprietary storage format used in DIVA Core releases 1.0 through 6.5.1.

See also *Media Format*.

Manager

The Manager is the DIVA component of the archive also hosting the archive system database.

See Manager in the DIVA Installation and Administration Guide.

Manager's Database

This is the postgres (formerly Oracle) database used by DIVA.

See also *DIVA Database*.

MDF (Metadata File and .mdf)

The term MDF is an acronym for the Drop Folder Monitor Metadata File. The MDF is a file containing information about the files included in a File Set, and has the file extension .mdf.

Measurement

A reading of specific information from an event or a resource. For example, the duration of a disk write operation, or the occurrence of a read error on a tape drive.

Media Format

Tapes and disks can be formatted as either AXF or Legacy (the format used before release 7.0) format. The format is set for tape groups and disk arrays during configuration. Complex Objects must be stored on AXF-formatted media.

Medium (Media)

Set of storage resources. DIVA provides two types of media: groups of tapes and arrays of disks. `DIVA_archiveObject()` and `DIVA_copyToGroup()` jobs transfer to a Medium (media). Also, storage media accessible to SPM (disk arrays and tape groups).

Metadata Database

The metadata database is the location where the metadata for components of Complex Objects are stored in the DIVA system.

Operations to save and read date to and from the Metadata Database are done via the MDS Metadata Service module.

Metadata Database for Complex Objects

To effectively operate with large volumes of files and folders and other metadata, DIVA stores the metadata separately from the Postgres database in the Core Metadata Database (*MongoDB*). The Core metadata database contains files stored in a file system local to DIVA. The directory that contains these files is the Metadata Database Root Folder.

The metadata database has very high performance and almost unlimited scalability. The metadata database should be treated with the same caution as the Postgres database, and should be backed up at regular intervals using the DIVA Backup Service (BKS).

Metadata File

The file listing the Object Name and Object Collection contained on a tape and its location.

Metadata Service

Tool for configuring the Metadata Database.

See the *Metadata Database* in the *DIVA Installation and Administration Guide*.

Migration Service

The Migration Service module creates migration jobs for DIVA, and schedules them for processing.

See *Migration Service Module* in the *DIVA Installation and Administration Guide*.

Metasources

The Metasource source type allows several currently-defined DIVA servers sharing the same online storage (or monitoring the same folder or FTP server for Drop Folder Monitors) to be combined and considered a single DIVA server configuration. This unique (and optional) feature enables DIVA to provide automatic load balancing and fail-over capabilities in case of one or more of the individual servers going offline.

Migration

Action of Copying or Moving of Object Instances stored on media of a given type to media of another type or another technology.

MPIO (Multipath I/O)

MPIO (Multipath I/O) is a Microsoft-provided framework that allows storage providers to develop multi-path solutions that contain the hardware-specific information needed to optimize connectivity with their storage arrays.

MultiDiva Mode

When an API client is connected to DIVA Connect in MultiDiva mode, multiple DIVA sites appear to the client as one large archive system.

Mutex (Mutual Exclusion)

Mutual Exclusion avoids the simultaneous use of a common resource (that is, mutual exclusion among threads).

Name

Part of the access key to an object. Names (file names) typically identify the object based on the content within the object.

NIC Teaming

The process of combining multiple network adapter cards together for performance and redundancy reasons. Microsoft refers to this as NIC Teaming, however other vendors may refer to this as bonding, balancing, or aggregation. The process is the same regardless of which solution is used or what it is called.

Non-complex Objects

Objects with less than 5,000 files are considered non-complex objects. The maximum number of files an object can hold is configurable.

Object

Each media file that is archived to DIVA is called an object.

An object is a DIVA logical container for all files consisting of a media file from the original source. Media files from some sources may have separate video, audio and metadata files. When archived in DIVA all of these files are referenced as a single object. When the object is restored to a destination, all files that were originally associated with that media file are automatically restored by DIVA.

An object is uniquely identified in DIVA by its name and Collection. The Object Name doesn't necessarily need to match that of the source file being archived. DIVA always restores the archived files as they were archived, regardless of the DIVA Object Name. Therefore, the same source file can be archived more than once in the same Collection, if each instance has a unique Object Name.

After an object exists within DIVA, it can't be replaced unless it is first deleted. If an Archive job uses the same name and Collection of an existing object, DIVA automatically stops and aborts the job. However, multiple copies (or instances) of an object can be created after the media file is archived.

If a source media file is to be stored in a variety of encoding formats (for example, MPEG2 Long GOP, DV50, or low resolution proxies), you can use specific categories to archive the same object based on its encoding format.

Object Instances

Mapping of an object's components onto a set of storage resources belonging to the same storage space. Deleting instances can't result in deleting the related object. The deletion of a unique instance is not permitted.

The storage managed by DIVA falls into three distinct categories:

- Online Storage (tapes within a library)
- Nearline Storage (disks and cloud storage)
- Offline Storage (externalized tapes)

The Name and Collection of an object in DIVA must be unique. However, multiple copies of that object can be created in one or all three of the above classes. Each copy of an object (including the original archived object itself) is known as an object instance.

Apart from creating backup copies, the object instances concept also allows life cycling of material within DIVA. An object may initially be created in online storage for rapid access and also backup instances created on one or more tapes. When the object is no longer required for online or "nearline* access, the disk instance can be deleted and the tape externalized. Automatic life cycling of objects, based on their age and location within the archive, can be provided by the SPM (Storage Policy Manager) option.

The first instance of an object is created when it is first archived to DIVA. Additional instances of the archived object can then be created with the Copy command.

An additional instance of an object can't be created by re-archiving the original object with the same name and Collection. This job is automatically stopped by DIVA with the "Object already exists within DIVA Core" error.

Instances are initially numbered sequentially with the original object that is archived to DIVA being Instance 0. As new instances are created and older instances deleted, the instance numbering may no longer be sequential when an object's properties are viewed in the DIVA Web App on the *Content Management* > *Catalog Browsing* view. However, an instance number from a previously deleted instance may be subsequently reused by DIVA in additional copy jobs.

The following restrictions apply to creating new instances of an object in DIVA:

- A tape group can contain two instances of the same object if both can be located on separate tapes. If no additional tapes for that tape group are available to store the second instance the copy job is terminated.
- A disk array can contain two instances of the same object if both can be located on separate disks within the array. If no additional disk is available, the copy job is terminated.
- A cloud disk can contain one instance of a object.

When an object has multiple instances within the archive and a restore job is issued, DIVA performs as follows:

- If no instance number is specified in the job, DIVA chooses the instance that allows the job to complete in the shortest possible time. A disk instance is preferred over a tape instance. However, a tape instance may be selected in some configurations if the QOS specified in the job is Cache Only or Cache and Direct.
- If no instance number is specified in the restore job and a disk instance exists, but the disk is offline, the tape instance is automatically selected.
- If two or more instances are present on tape and no disk instances exist, and one tape is currently in use by another job (or is externalized), the tape containing the other instance is automatically selected.
- If two or more instances exist on tape, and a read error occurs on the first instance selected, the job is automatically attempted on the other instances until the job is completed successfully. If no instances can be read, the job is terminated.
- If a specific Instance Number is specified in the restore job, DIVA only uses that instance. If the media containing the instance is offline (for disks), externalized (for tapes), or an I/O or read error occurs, the job is terminated.

OU (Organizational Unit)

An Organizational Unit is a subdivision within an Active Directory into which you can place users, Tape groups, computers, and other organizational units. You can create organizational units to mirror your organization's functional or business structure. Each domain can implement its own organizational unit hierarchy. If your organization contains several domains, you can create organizational unit structures in each domain that are independent of the structures in the other domains.

Postgres Database

The third party database used for the Archive System Database.

Proxy Actor

Actors make all types of data transfers, such as Archive, Restore, Copy, and Repack.

See *Actors and Proxy Actors* in the *DIVA Installation and Administration Guide*.

Proxy Service

Running on Windows, the Proxy Service Module generates and manages additional metadata files. These include proxy files, thumbnails, and technical metadata files that you can extract from a transcoder engine.

See Proxy Service Module in the *DIVA Installation and Administration Guide*.

Quality of Service

The QOS (Quality of Service) parameter defines how a file is transferred to and from a tape, from a source, or to a target.

See also *Archive Submissions*.

Repack

Elimination of blank blocks between two objects on a tape (these blocks are caused by the deletion of objects), by moving the objects to a different, empty tape.

Resource

Used to denote the necessary elements involved for processing jobs (for example, Actors, disk, drive, and tape). A resource is a uniquely identified element of the DIVA system. Analytics App references them by events and metrics. The following are examples of resources in DIVA:

- Tape with Barcode ABE6785
- Actor Actor01
- Tape Drive Serial Number 134001021

Resource Type

These are generic types of resources. For example, tapes, tape drives, jobs, or objects.

Robot Core

The mechanical tape system used with DIVA to insert and eject tapes to and from the tape library.

Robot Manager

The Robot Manager module provides an intermediate software layer for DIVA to interact with many different types of tape Managed Storage.

See *Robot Manager* in the *DIVA Installation and Administration Guide*.

SAS (Serial Attached SCSI)

A point-to-point serial protocol that moves data to and from computer storage devices such as hard drives and tape drives.

Server

Contains information necessary to communicate with a connected server or disk. You transfer content to and from DIVA through these servers and disks. Examples are Video servers, FTP servers, and disk arrays.

Set (of tapes)

Every tape in a DIVA system belongs to one and only one Set. If the tape is not available to DIVA, it belongs to Set #0, otherwise it belongs to a set with a strictly positive ID (for example, Set #1). Each Tape group is associated with a Set. When the tape group needs an additional tape, it takes it from its associated Set.

See also *Watch Folder (Drop Folder)*.

Site-to-Site Mapping

Describes how DIVA Connect will copy objects from one site to another.

Slot

Contains the action to be applied when the associated Storage Plan is executed.

Source

A system that produces data to be archived in the DIVA system (for example, video servers, browsing servers, remote computers, and so on). Source servers can also be used as a Target (Destination) for certain operations.

See also *Watch Folder (Drop Folder)*.

Spanning

Splitting an object's component onto several tapes (usually two). This can occur when the component size is larger than the remaining size left on the initial tape.

SSL (Secure Sockets Layer)

A Secure Sockets Layer is a standard security protocol for establishing an encrypted connection between a server and a client. Specifically, it encrypts the connection and the data transmitted along the connection. To achieve a secure connection, a service needs a Key Pair (Public Key and Private Key) and SSL Certificate.

SSL Certificate Authentication

An SSL certificate is a digital certificate that authenticates a service in network connections. To generate an SSL certificate, you must create a CSR (Certificate Signing Request) for your service Key Pairs and have it signed by your CA (Certificate Authority). An SSL certificate contains the following information:

- Certificate holder's name
- Certificate serial number and expiration date
- A copy of the certificate holder's public key
- Digital signature of the certificate issuing authority

SSL Certificate Chain

There are two types of CAs (Certificate Authorities): Root CAs and Intermediate CAs.

A certificate chain is an ordered list of certificates, containing an SSL Certificate and Certificate Authority Certificates that enable the receiver to verify that the sender and all CAs are trustworthy using its trust store. The chain (or path) begins with the SSL certificate, and each certificate in the chain is signed by the entity identified by the next certificate in the chain. Any certificate that sits between the SSL Certificate and the Root CA Certificate (last certificate in the chain) is called an Intermediate CA Certificate. The Root CA is at the end of the chain and it signs the intermediate CA certificate, and the Intermediate CA signs the SSL certificate for the services.

For example, when a service receives its peer's SSL certificate chain that is trying to connect during the SSL handshake process, it verifies its peers SSL certificate in the chain using the Intermediate CA certificate next in the chain. It then verifies the Intermediate CA certificate by looking for the Root CA certificate that signed the intermediate CA certificate in its trust store. This verification completes the Certificate Chain. Connection is not established if the full chain verification fails.

Step

The current state of progression of an action or process.

Storage Media Format

In DIVA, 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 DIVA Legacy format or one of the AXF formats. This setting can be changed at any time and it doesn't influence content already stored.

- Tape Storage Media Format

  Although a tape group can contain more than one storage format, an individual tape has (at most) one storage media format. DIVA assigns the tape media format to an empty tape when it writes the first object to that tape.

- Disk Storage Media Format

  Unlike tapes, disks do not have a format. DIVA allows storing objects in different media formats on the same disk.

- Object Instances Media Format

  Every tape and disk object instance is assigned a format of Legacy or AXF. The format of a tape or disk instance is assigned when the instance is created and is the format of the tape on which the instance resides. All instances on a tape must have the same format.

Storage Plan

Actions to execute when new content arrives.

Storage Policy Management (SPM)

Storage Policy Manager software component enables object life-cycle (interacting with the Manager) management, and is typically installed on the same system as the Manager. For example, an archived object can reside on a specific medium the first day, and migrate (over time) to a different medium according to the policies and rules established by the user. Manager executes the object life-cycle migration as a

background activity by following the rules and policies defined in the corresponding Storage Plan.

SNMP Agent (optional)

The SNMP (Simple Network Management Protocol) Agent and MIB (Management Information Base) supports status and activity monitoring of DIVA and its subsystems to a third party monitoring application through the SNMP protocol.

See SNMP Agent in the *DIVA Installation and Administration Guide*.

Storage Policy Management (SPM)

The SPM (Storage Policy Manager) software component provides object life cycle management (interacting with DIVA), and is typically installed on the same computer as DIVA.

See Storage Policy Manager (SPM) in the *DIVA Installation and Administration Guide*.

Tape Groups and Sets

A tape group is a logical notion for characterizing a set of object instances. This concept has a direct influence on the instance's storage policy for tapes. Instances of the same tape group will be stored on the same tapes. However, objects can't have multiple instances stored on the same tape.

Disks are logically assigned to arrays for the storage of objects, but tapes are logically associated together in tape groups. Tape groups include the following functionality:

- Sony ODA drives are supported
- Tape Compression
- Tape Group Encryption
- Tape Spanning
- Protected Mode
- Tape Label Management

Tape groups are based on the DIVA Tape Set. Each tape inserted in the system is assigned to a Set. Tape groups are then associated with a single Set. Multiple tape groups may be associated with the same set. No tape group can use the set number 0.

Several kinds of tape can be used in a DIVA system. Tape groups can be defined either by using a Set, in which you assign only tapes of the same type, or by defining the Set in which you can mix tape types. Therefore, the first case specifies the tape type that stores the object instance.

For more information, see Set (of tapes).

Target (Destination)

A system that receives restored data in the DIVA system (for example, video servers, remote computers, FTP servers, and so on). Target servers can also be used as a source for certain operations.

Transcoder Support

The DIVA Actor can integrate with a transcoder engine to provide real-time transcoding of material as it is archived or restored, or to create objects from already existing

content within the archive. DIVA supports integration to Telestream Vantage. Multiple transcoders are supported for Vantage.

Transfer

Copying data from a source to a DIVA media (Archive job) or from a DIVA media to a target (Restore job).

For more information, see Jobs.

TRU (Tape Reading Utility)

Used in DIVA releases 6.5 and earlier to read tapes; the TRU tool remains in later versions of DIVA for support and investigating problems.

Trust Store

A Trust Store contains the certificates of CAs (Certification Authorities) you trust. For example, when a service receives its peer's SSL certificate that is trying to connect during SSL handshake process, it verifies that its peer's SSL certificate's digital signature is signed by one of the trusted certificates in its trust store. If the certificate is not in the Trust Store, the SSL handshake fails and the connection is not established.

Unmanaged Storage

Source or destination servers on which the material to archive into or restore from DIVA is located. Examples are: broadcast video servers, video editing systems, FTP sites, and generic computer systems.

UUID (Universally Unique Identifier)

A Universally Unique Identifier uniquely identifies each object created in DIVA across all Telestream customer sites except for objects created through Copy As jobs. An object created using a Copy As job will contain the same UUID as that of the source object.

VACP Converter (optional)

The Video Archive Communications Protocol is developed by Harris Automation Solutions and used by some automation systems for interfacing to an archive system.

See Video Archive Communications Protocol (VACP) in the *DIVA Installation and Administration Guide*.

Watch Folder (Drop Folder)

A folder on a local disk, FTP server, or a CIFS shared folder designated for Single File mode, File Set mode, or both, storage that will be monitored by Drop Folder Monitor and from which files will have operations performed on them.

Web App

Web based interface used to configure and monitor a DIVA system. This replaces the legacy java-based Configuration Utility and Control GUI, which are deprecated starting from DIVA 9.1.

Workflow Profile

A set of parameters that defines how incoming jobs will be processed by DIVA Connect. Multiple profiles can be created and mapped to input ports.

Workflow Engine Service

The Workflow Initiator service enables a robust integration between DIVA and Telestream Vantage, and between DIVA and a large variety of encoders.

WS API (deprecated in favor of DIVA REST API)

Telestream has deprecated WS API in favor of DIVA REST API.

See DIVA REST API in the DIVA Installation and Administration Guide.