



Telestream



Content Conductor

Storage Policy Manager User Guide

Release: 9.0

Revision: 1.0

Copyrights and Trademark Notices

Specifications subject to change without notice. Copyright © 2023 Telestream, LLC and its Affiliates. Telestream, CaptionMaker, Cerify, DIVA, Content Conductor, Episode, Flip4Mac, FlipFactory, Flip Player, Gameshow, GraphicsFactory, Kumulate, Lightspeed, MetaFlip, Post Producer, Prism, ScreenFlow, Split-and-Stitch, Switch, Tempo, TrafficManager, Vantage, VOD Producer, and Wirecast are registered trademarks and Aurora, ContentAgent, Cricket, e-Captioning, Inspector, iQ, iVMS, iVMS ASM, MacCaption, Pipeline, Sentry, Surveyor, Vantage Cloud Port, CaptureVU, Cerify, FlexVU, PRISM, Sentry, Stay Genlock, Aurora, and Vidchecker are trademarks of Telestream, LLC and its Affiliates. All other trademarks are the property of their respective owners.

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

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

Contents

Telestream Contact Information 6

Preface 7

- Audience 8
- Documentation Accessibility 8
- Related Documents 8
- Document Updates 8

Overview 9

- Storage Policy Manager Overview 10
- New and Enhanced SPM Features and Functionality 10
- Object Life Cycle 10
- Managing the Object Life Cycle 11
- SPM Slot Types 11
 - Actions, Action Steps, and Action States 12
- Managing Watermarks and Arrays 14
 - Recommended Best Practices for Watermarking 15
- Storage Policy Manager Workflows 18
 - Storage Policy Manager Tasks 18
 - Simple Object Life Cycle Example 19
 - Retrying SPM Failed Actions and Issue Resolution 19
 - Action Slot Workflow 21
 - Storage Slot Workflow 21
- Recommended Best Practices 22
 - SPM Configuration 22
 - Choosing Appropriate Watermarks 23
 - General Watermarking Rules 24
 - Simple SPM Configuration Example 24
 - Additional SPM Configuration Examples 25

Installation and Configuration 27

- Installation Prerequisites 28
- SPM Configuration File Parameters 28
- Request Type Distribution 33
- Storage Plan Definition 35
 - Storage Plan Settings 35
 - Recommended Best Practices 35
 - Creating the Storage Plan 35
 - Creating Mediums 36
 - Creating Filters 38
 - Alternate Methods of Assigning Storage Plans to an Object 40
- Creating Slots 41
 - Configuring Storage Slots 41
 - Configuring Nearline Storage Slots 45
 - Configuring Transcode Archived Slots 47
 - Configuring Metadata Archive Slots 53
 - Configuring Restore Slots 57
 - SPM Delete Object Behavior 62
 - ALLOW_OBJECT_DELETION Parameter 62
 - DELETE_OBJECT_ONLY_LAST_INSTANCE Parameter 63
- Changing a Storage Plan 64
 - Disabling and Deleting Slots 65
 - Adding a Slot 65
- Watermark Based Disk Cleaning Management 66
- External Storage Plans 67
- Configuration Validation 67

Operations 69

- Managing the SPM Service using the Command Line 70
- Accessing SPM Information using the Web App 71
 - Filtering the SPM Actions Page View 71
 - Assigning a New Storage Plan 72
 - Rescheduling Failed Actions 72
 - Marking Failed Actions Complete 73
- Storage Policy Manager Action Result Codes 73
- Storage Policy Manager Logging 76

Frequently Asked Questions 77

Default Configuration Files and Parameters 80

- SPM Configuration Parameters 81
 - Storage Plan Definitions 81
 - Mediums Definitions 81
 - Object Slot Definitions 82

Filter Definitions	83
SPM Actions Definitions	84
Object Filters	84
Request Templates	85
SPM Default Configuration File (SPM.settings.json.ini)	86

Telestream Contact Information

To obtain product information, technical support, or provide comments on this guide, contact us using our web site, email, or phone number as listed below.

Resource	Contact Information
Content Conductor Technical Support	Web Site: https://www.telestream.net/telestream-support/content-conductor/support.htm Depending on the problem severity, we will respond to your request within 24 business hours. For P1, we will respond within 1 hour. Please see the Maintenance & Support Guide for these definitions. • Support hours for customers are Monday-Friday, 7am-6pm local time. • P1 issues for customers are 24/7.
Telestream, LLC	Web Site: www.telestream.net Sales and Marketing Email: info@telestream.net Telestream, LLC 848 Gold Flat Road, Suite 1 Nevada City, CA USA 95959
International Distributor Support	Web Site: www.telestream.net See the Telestream Web site for your regional authorized Telestream distributor.
Telestream Technical Writers	Email: techwriter@telestream.net Share comments about this or other Telestream documents.

Preface

This document describes the installation, configuration, and operations of the Content Conductor Storage Policy Manager.

Topics

- [Audience](#)
- [Documentation Accessibility](#)
- [Related Documents](#)
- [Document Updates](#)

Audience

This document is intended for Installation, Administration and Operations personnel to follow all of the necessary steps to achieve full functionality of the SPM (Storage Policy Manager) component.

Documentation Accessibility

For information about Telestream's commitment to accessibility, visit the Telestream Support Portal located at <https://www.telestream.net/telestream-support/>.

Related Documents

For more information, see the Content Conductor library for this release located at: <https://www.telestream.net/telestream-support/content-conductor/support.htm>

Document Updates

The following table identifies updates made to this document.

Date	Update
April 2022	Updated book for release 8.2. Updated terminology to new standards.
July 2022	Migrated book to Telestream format and styles.
September 2022	Updated terminology and title page graphic.
October 2022	Updated book for release 8.3
November 2022	Reverted the term "Virtual Object" to "Object".
December 2022	Updated book for 8.3.1 release.
February 2023	Updated book for 9.0 release.
May 2023	Updated spm.conf.ini to SPM.settings.json.ini. Remove references to spm.trace.ini (now included in SPM.settings.json.ini). Updated web app terms.
June 2023	Updated book with comments received and sent for final review.
September 2023	Change product name to Content Conductor.

Overview

This chapter describes an overview of the Storage Policy Manager, and includes the following information:

Topics

- [Storage Policy Manager Overview](#)
- [New and Enhanced SPM Features and Functionality](#)
- [Object Life Cycle](#)
- [Managing the Object Life Cycle](#)
- [SPM Slot Types](#)
- [Managing Watermarks and Arrays](#)
- [Storage Policy Manager Workflows](#)
- [Recommended Best Practices](#)

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), or database corruption (for example, creating a Storage slot while SPM is running, and changing it to Restore type slot while copy and delete instance actions have already begun processing). 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 Content Conductor system or permanent data loss.

Storage Policy Manager Overview

The Storage Policy Manager software component provides object life cycle management (interacting with Content Conductor), and is typically installed on the same computer as Content Conductor. 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. Content Conductor executes the object life cycle migration as a background activity by following the rules and policies defined in the corresponding Storage Plan.

New and Enhanced SPM Features and Functionality

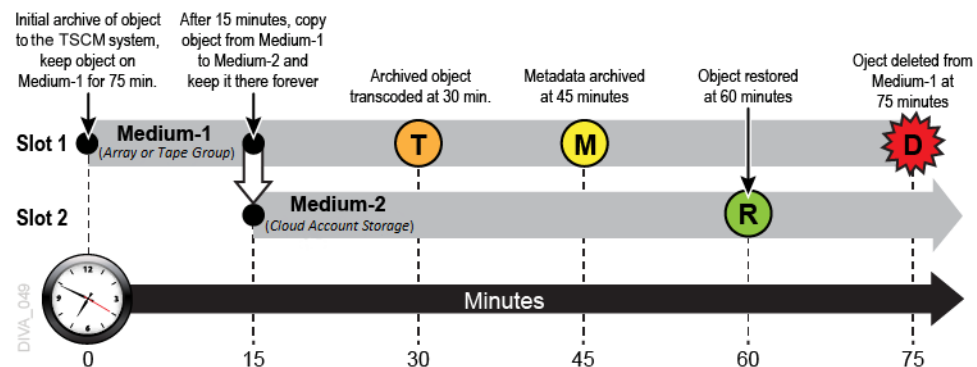
See the Content Conductor Installation and Configuration Guide and the Content Conductor Release Notes for more information on Content Conductor new and enhanced features and functionality.

Note: The `spm.conf` configuration file has been replaced with `SPM.settings.json`.

Object Life Cycle

This section describes the steps and processes encompassing the Content Conductor Object life cycle. Along with the standard Content Conductor terminology, Storage Policy Manager has additional terms that require identification. Understanding these additional terms will assist in comprehending the Content Conductor SPM module, the processes involved, and the Content Conductor Object life cycle.

There are several steps included in the object life cycle, depicted in the following example figure. SPM can be used to automate this generic object life cycle.



Managing the Object Life Cycle

Content Conductor 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 Content Conductor 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 called 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.

To create a slot, navigate to the Configuration > Resources > Storage Plans > Slots page in 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 page in 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 web app Configuration > Resources > Storage Plans > Slots page.

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 web app Configuration > Resources > Storage Plans > Slots page.

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 archiving of the object metadata. Create and configure this type of slot using the web app Configuration > Resources > Storage Plans > Slots page.

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 web app Configuration > Resources > Storage Plans > Slots page.

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 Archived, 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 screen, and all Slot Types only require configuration from the web app Slots tab.

When an object is added to Content Conductor, 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. See [Storage Policy Manager Workflows](#) for additional Slot Workflow information.

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.

Managing Watermarks and Arrays

You can manage deletions on disk mediums using watermarks. When watermarks are not 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.

See [SPM Configuration File Parameters](#) and [SPM Default Configuration File \(SPM.settings.json.ini\)](#) for configuration information.

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 is reached.

Objects on watermarked arrays are deleted using one of the two following 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 Content Conductor API. Requests are sent to the Content Conductor and the watermark levels are sent back to DSM.

SPM arrays have three possible Watermark settings that are configurable in the web app:

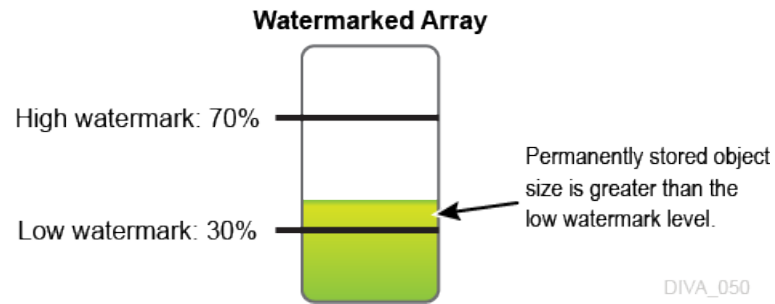
- Yes
This setting watermarks the identified array, and only considers objects already marked for deletion.
- No
This setting does not watermark the identified array.
- Mixed
This setting incorporates a combination of watermarked and not watermarked. This setting is only valid for Storage Slots. The 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.

Recommended Best Practices for Watermarking

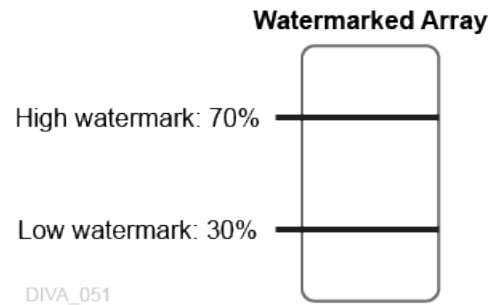
If the total size of all permanently stored objects (that is, objects with no specified expiration time) on an array is greater than the configured Low Watermark, Telestream recommends reconfiguring the Low Watermark setting to a value higher than the amount of permanently stored objects.

The following figure represents this basic concept:

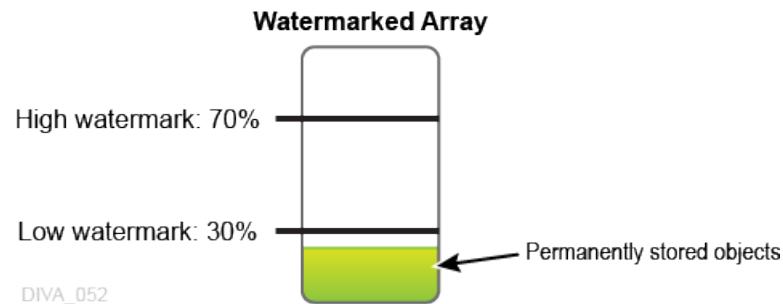


The following figures characterize a typical Watermarked Medium (watermarking set to Yes) and illustrate how watermarking ensures the medium does not reach full capacity.

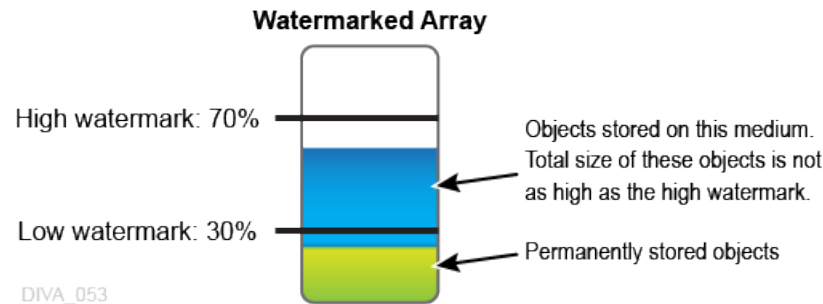
The following figure represents a watermarked disk array with no objects, and shows the Low Watermark and High Watermark:



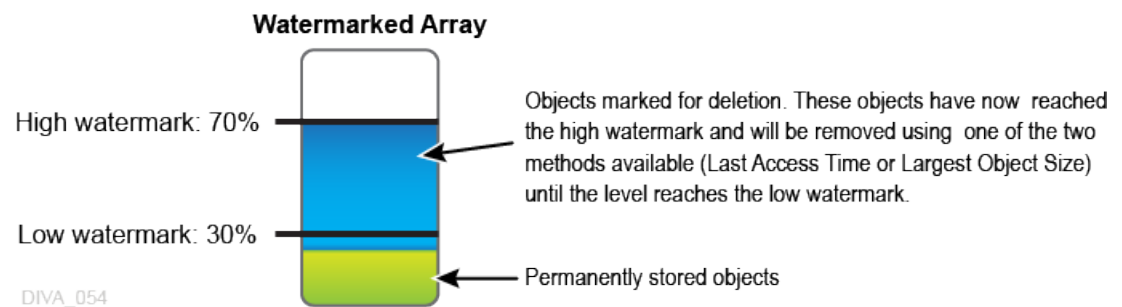
The following figure represents a watermarked disk array with permanently stored objects with a total size that is less than the Low Watermark:



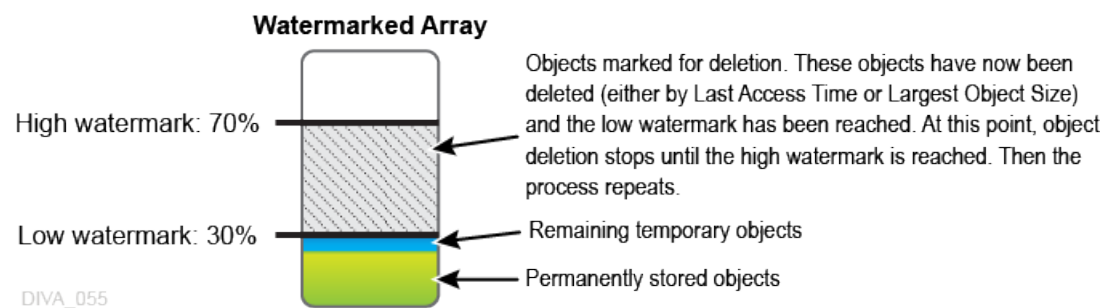
The following figure represents a watermarked disk array with permanently stored objects with a total size that is less than the Low Watermark, and temporarily stored objects with a total size that is higher than the Low Watermark:



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 Content Conductor 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 web app SPM Actions panel. The web app must at least be connected to the Content Conductor 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 Content Conductor as necessary.
- **DSM (Disk Space Monitor)**
DSM monitors a particular array (not individual disks) to make sure it does not 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 Content Conductor 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 Content Conductor.
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 Content Conductor
 - 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 Content Conductor, 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 Content Conductor system and SPM 10 minutes later (20 minutes after initially being archived to Content Conductor).

Retrying SPM Failed Actions 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.

- Content Conductor Connection Failures

SPM uses the Content Conductor API to connect to Content Conductor. If the Content Conductor connection is down, SPM continues retrying to establish the connection to Content Conductor until a connection is established.

- Database Connection Failures

SPM will continue retrying to connect to the Content Conductor 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 does not 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 does not 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. See [Action Slot Workflow](#) and [Storage Slot Workflow](#) for Action Slot and Storage Slot workflows.

- Content Conductor is Busy

If SPM could not execute the actions by submitting a job because Content Conductor 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 Content Conductor 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

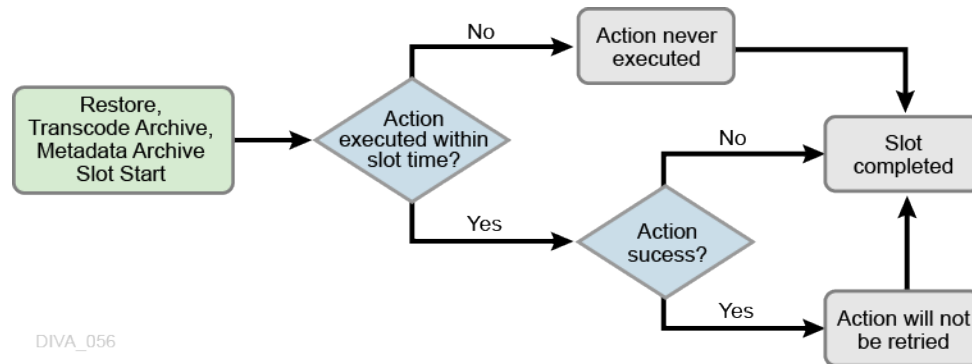
If the Once Only parameter is set to false for a Storage Slot, all of its actions will be retried throughout the Slot Start Time and Slot End Time period.

Action Slot Workflow

Action Slots have a specific workflow that encompasses what Content Conductor 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:

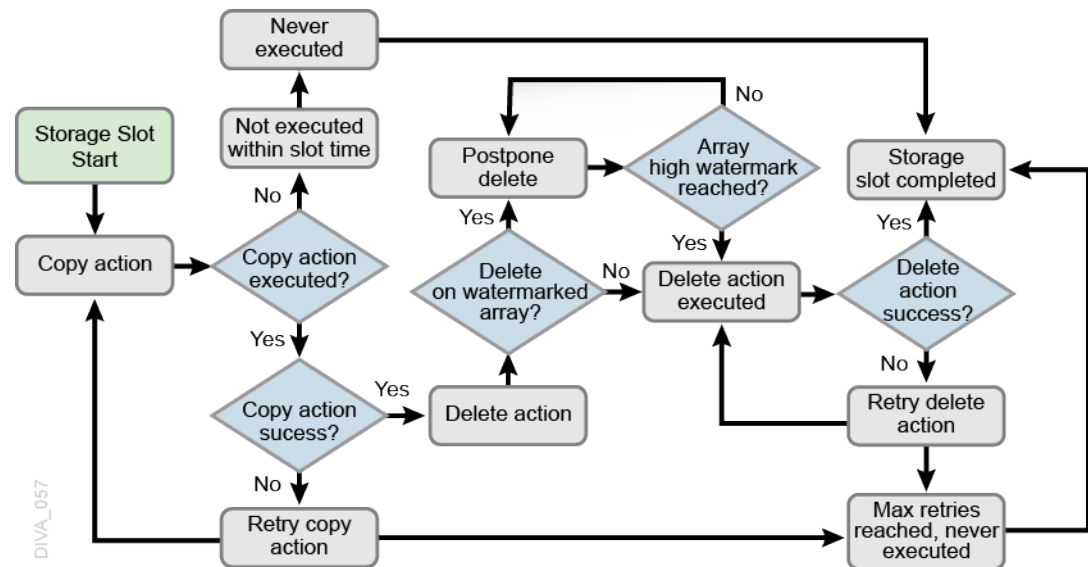


Storage Slot Workflow

Storage Slots also have a specific workflow that encompasses what Content Conductor 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:



Recommended 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.

See [Action Slot Workflow](#) and [Storage Slot Workflow](#) for Action Slot and Storage Slot workflows.

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 Content Conductor, and then updates the action to the Complete state.

See [Managing Watermarks and Arrays](#) for more information on watermarking 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 does not 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 is not 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).

See [Managing Watermarks and Arrays](#) for more information on watermarking 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, Content Conductor 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 Content Conductor 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.

See [Managing Watermarks and Arrays](#) for more information on watermarking 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 cannot be purged (for example, objects belonging to a Storage Plan that keeps disk instances for an unlimited time, or user files not belonging to Content Conductor), 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.

See [Managing Watermarks and Arrays](#) for more information on watermarking arrays.

Simple SPM Configuration Example

The following figure illustrates a simple SPM configuration example.



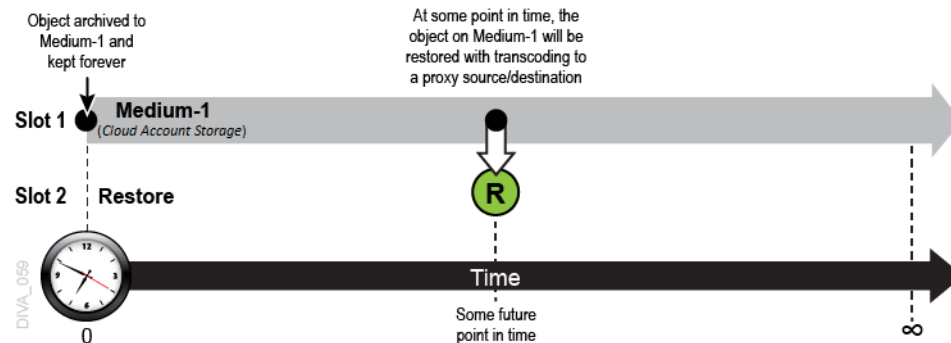
The following occurs in this scenario:

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 Content Conductor 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.

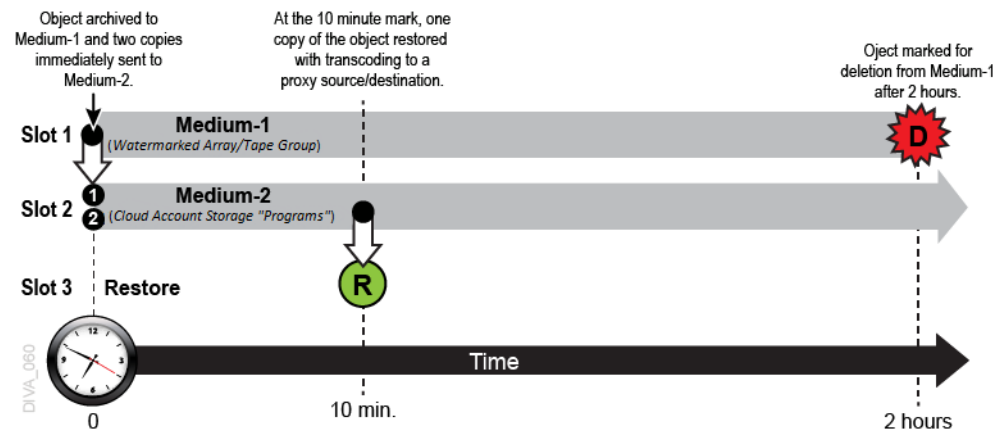
See [SPM Slot Types](#) and [Creating Slots](#) for more information about Slot Types and Slot configuration.

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.

The following 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 Content Conductor 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 Content Conductor Documentation Library on the Content Conductor Technical Support site for detailed information for full installation and configuration instructions for specific additional elements where necessary.

Topics

- [Installation Prerequisites](#)
- [SPM Configuration File Parameters](#)
- [Request Type Distribution](#)
- [Storage Plan Definition](#)
- [Creating Slots](#)
- [Changing a Storage Plan](#)
- [Watermark Based Disk Cleaning Management](#)
- [External Storage Plans](#)
- [Configuration Validation](#)

Installation Prerequisites

There are several prerequisites required before installing SPM as follows:

- The Content Conductor database must be Postgres 14 or later.
- The Content Conductor database user name and password must be the same ones used for the Content Conductor installation.
- Content Conductor must be installed and running.
- Content Conductor 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 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 web app.

The following parameters are for Manager:

- The following are connection parameters for the Content Conductor database:
 - User
This parameter identifies the Content Conductor database user name. (Default: none).
 - Password
This parameter identifies the password. (Default: none).
 - ConnectionString
This parameter identifies the Content Conductor database connection string. This is usually the SQL*Net address name declared in the local Postgres file for the Content Conductor 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 Content Conductor is currently processing 75 jobs total, and only 25 of them are Delete jobs, SPM will send another 25 Delete jobs to Content Conductor, because both numbers are less than 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 cannot 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 Archive 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 Content Conductor 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 Content Conductor 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. Current releases of Content Conductor SPM is more dynamic in nature than previous releases, 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 Content Conductor 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 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 will be 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 Content Conductor 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 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 parameters 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.

If the monitored array is a password protected network share, the following syntax is allowed as a mount point:

`cifs://user:pwd@\\nas\share`

When the SPM Service is configured with a CIFS array, the log in of the SPM Service must be changed from the default Local System to a valid Windows user.

UNC paths are supported for SMB managed disks if the UNC path is directly mounted on the Windows Actors.

Use the following process to change the SPM Service login:

1. Run the Windows Services application (services.msc).
2. Select the SPM Service requiring the log in change.
3. Right-click the SPM Service, and then select Properties.
4. Select the Log On tab in the Properties window of the SPM Service.
5. Select the This account option.
6. Enter a valid Windows user name and password.
7. Click Apply (or OK) to commit the changes.

Examples:

`ARRAY1 = ShortClips;75;90` (this is the recommended syntax)

`ARRAY2 = Raid_001;60;85;X:\;Y:\;Z:\` (this is the legacy syntax)

- The following is a SPM.settings.json Log level parameter:

- `LogLevel`

This parameter identifies the level of tracing to use for the logs. Refer to the [SPM Default Configuration File \(SPM.settings.json.ini\)](#) section for detailed settings.

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 cannot 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 web app.

There are two tabs in the web app related to SPM configuration; Storage Plans and Slots.

Storage Plan Settings

The Storage Plan Settings in the web app 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 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 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 Content Conductor 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 Content Conductor 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 Content Conductor Automatic Data Migration from Avalon to Content Conductor.

- Tape Group/Array Name

Select the default media from the list of media managed in the Content Conductor system. Content Conductor 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 Content Conductor 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 web app.

Creating Mediums

The Media Groups Section has two purposes:

- Any medium that is defined in the Media Groups section of the web app Storage Plans tab is available for use by SPM. If a new object is placed on a medium not included in the Media Groups list, it will not be managed by SPM and is assigned the `SPM_DEFAULT` Storage Plan.
- To select a medium as the target for actions when creating slots, it must be defined in the Media Groups list.

Use the following procedure to create a Media Group:

1. Navigate to the web app Storage Plan settings.
2. Click the + icon on the top of the Media Groups section.
3. When the Add New Row dialog box appears, begin configuring the new mediums.

4. Enter the information in the appropriate fields, and use the menu lists to select the remaining options as follows:

– Medium Name

Enter the Medium Name in the Medium Name field.

– Storage Name

Enter the desired Storage Name in the Storage Name field. The recommended Storage Name is DIVA.

– Tape Group/Array Name

Select the Tape Group or Disk Array to be associated with this medium from the list.

The selections available in the list are determined by what media is defined in the Content Conductor system.

– Watermarked

Monitors the Medium's usage (available space, used space, and so on). Select **Y**, **N**, or **M** to specify whether to apply watermarking to this medium. Refer to Sections [Choosing Appropriate Watermarks](#) and [General Watermarking Rules](#) for detailed information on Watermarking.

In general it is recommended that slots are not deleted, but disabled instead. Deleting a slot does not remove the content from the Content Conductor system.

In the case of a Storage Slot, setting the Slot End Time to 0 will force SPM to delete any contents created by this slot and once all the contents are removed, the Storage Slot can be deleted or disabled. If retaining the content in the Storage Slot is desired, but no further processing of future actions is required, disabling the slot instead of deleting it is the safer way to do this.

The Watermark-Based Disk Cleaning Management options are as follows:

– Yes (Y)

This selection applies watermarking.

– No (N)

This selection does not apply watermarking.

– Mixed (M)

This selection applies a combination of watermarking. The disk cleaning action taken depends on what event occurs first; either the slot reaches its End Time, or the High Watermark is reached.

Mixed Mode only works if Once Only is set to **Y** (yes) for the slot.

Watermarks are only used for Storage Slots, and only for disk arrays configured as Mediums.

– Disk Cleaning Strategy

Defines the cleaning strategy to perform on objects marked for deletion on the Array when the high watermark is reached. See [Watermark Based Disk Cleaning Management](#) for detailed information on disk cleaning management.

There are two options available:

- * By Last Access Time
Older objects are cleaned up before recently accessed objects.
- * By Object Size
The largest objects are cleaned up first until the Low Watermark is reached.

5. Click OK to complete the process.

The new Media Group is now displayed in the Media Groups section of the Storage Plans tab.

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.

The following figure shows filters that are matches and are optional to alter. To get them to work, you cannot archive using a storage plan or it bypasses the filters.

When a new object is archived into Content Conductor, and it matches a defined filter, a Storage Plan is assigned to that object as follows:

- If the new object does not 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.

Use the following procedure to create a filter:

1. Navigate to the 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 page.

Alternate Methods of 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 web app.
- Using the Content Conductor REST API. See the REST API documentation on the Content Conductor Technical Support site.
- 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.

Configure Media Mapping in web app by navigating to Configuration > Resources > Cloud Storage > Cloud Buckets, or Configuration > Resources > Disk Storage > Arrays as follows:

1. Open web app and navigate to the appropriate section under the Configuration menu item.
2. Click the + icon to open the Add new row in Media Mapping dialog box.
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.

Creating 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. See [Watermark Based Disk Cleaning Management](#) for watermarking details.

The copy action will occur only if the slot target medium does not 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).

Use the following procedure to configure a Storage Slot:

1. Open the web app, and navigate to the Configuration > Resources > Storage Plans > Slots page.
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 are not 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 Content Conductor.

A DeleteInstance Action reaching its execution time will not send the Delete Instance job to Content Conductor, but if somebody manually deletes that instance using the 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 will allow for 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 does not 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 page in the web app.

Configuring Nearline Storage Slots

Two additional Quality of Service (QOS) parameters called NEARLINE_ONLY and NEARLINE_AND_DIRECT can be specified through the API or the 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 Content Conductor 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 a Content Conductor Actor-Disk connection that is configured for Nearline. If no disk instance is found, Content Conductor 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 will terminate and an error is generated stating 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 not 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 Content Conductor 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.

Content Conductor 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 web app stating Nearline and Nearline & Direct QOS are not 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 Content Conductor 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 are not 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 page in the web app and contain only a single action. This slot will take an existing clip in a Content Conductor 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 the Content Conductor System, 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. Linux-based Actors only support Telestream Vantage transcoding operations.

Use the following procedure to configure a Transcode Archived Slot:

1. Open the web app and navigate to the Configuration > Resources > Storage Plans > Slots page.
2. Click the + icon in the Slots section of the page.

3. When the Slot Configuration window opens begin configuring the new slot by selecting `TRANSCODE_ARCHIVED` from the Slot Request Type list, and then continue with the remaining configuration parameters 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 `TRANSCODE_ARCHIVED` 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 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. Selecting Y indicates the slot is enabled, and N indicates it is disabled. When a slot is disabled its actions are not executed, however they are still created, and the status is updated. When the slot is enabled again, pending actions are executed immediately.
- 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 will allow for 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 Content Conductor.

- 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.

4. Click OK to complete the process. The new slot is displayed on the Slots page under the Configuration > Resources > Storage Plans > Slots page.

There is a second set of buttons at the bottom of the Slot Configuration page that contains the following:

- 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.

5. Using the Restore, Transcode & Metadata Archive fields on the Slot Configuration page, 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 are not 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.

6. Similar to the Transformation Rules, the Request Templates parameters are located at the bottom of the Slot Configuration page. 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 (currently `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, Content Conductor 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, Content Conductor 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 web app. The available options are as follows:
 - * `-tr_names`
This option specifies the transcoder to use as defined in the 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 Content Conductor 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 the Metadata Archive Slots on the Slots page of the web app. These slots contain only a single action. This slot will archive 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 location (Server and Path) that is defined in the Slot Configuration window. The Object Name and Object Collection used for the metadata file archive are controlled by the Transformation Rule entries.

Use the following procedure to configure Metadata Archive Slots:

1. Open the web app, and navigate to the Configuration > Resources > Storage Plans > Slots page.

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 METADATA_ARCHIVE 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), and the Metadata Archive 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 the metadata file is archived 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, as soon as the slot is open the archive 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 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 are not executed, however they are still created when a new object is archived.

– **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 will allow for 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 metadata 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.

– Request priority

This field identifies the priority of the job and order of execution when the action is submitted to the Manager.

– Medium Associated with Slot

Select the target medium to associate with the slot from the list.

4. Similar to the Transformation Rules, the Request Templates parameters are located at the bottom of the Slot Configuration screen 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 are not 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 Content Conductor 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 (currently 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, Content Conductor 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, Content Conductor 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 Content Conductor 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 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 page under the Configuration > Resources > Storage Plans > Slots page.

Configuring Restore Slots

Configure Restore Slots on the Slots page of the web app. Restore Slots only contain a single action. This slot will take an existing Content Conductor object and restore it onto the designated Server.

Use the following procedure to configure Restore Slots:

1. Open the web app, and navigate to the Configuration > Resources > Storage Plans > Slots page.
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 RESTORE 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), 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 are not executed. However, they are still created when a new object is archived.

– Request 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).

– Request 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 will allow for 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.

– Request 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 screen 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 are not 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 Content Conductor 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 (currently 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, Content Conductor 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, Content Conductor 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 wild-card 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 in the Slots page under the Configuration > Resources > Storage Plans > Slots page.

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 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 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 Content Conductor 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 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 Content Conductor system or on the medium, only Delete Instance is executed.

- Delete Object is executed only if one instance exists on the entire Content Conductor 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 Watermarked, the Delete Object is postponed until the Watermark condition is satisfied.

Changing a Storage Plan

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 Content Conductor 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 Content Conductor 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:

1. Object1234 is archived to DISK1 under Storage Plan A.
2. Storage Plan A has slots for the following operations, and Object1234 is copied to both Tape Groups:
Copy to NEARLINE
Copy to OFFLINE

3. 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.
4. The new slot change is added that will copy Object1234 to DISK2 but it is necessary to also delete the object instance from the NEARLINE Tape Group.

To delete the object instance from the NEARLINE Tape Group, the Slot End Time must be changed to 0 for the NEARLINE Storage Slot (see [Disabling and Deleting Slots](#) for more information).

A new Storage Slot must be added on DISK2 with a Slot Start Time = 0 and Slot End Time = -1 (see [Adding a Slot](#) for more information).

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.

Disabling and Deleting Slots

Telestream generally recommends that slots are not deleted, but rather disabled instead. Deleting a slot does not remove the content from the Content Conductor system.

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.

Adding a Slot

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 (see [Configuring Storage Slots](#) for valid parameters). 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.

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 Content Conductor 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

Content Conductor 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

Content Conductor 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 does not 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 web app.

External Storage Plans

External Storage Plan functionality allows management of the Content Conductor Object life cycle using third party external systems. The SPM Database API can be used to develop integration of third party components with Content Conductor and SPM.

Note: SPM, E (External) should only be used by Telestream personnel.

Configuration Validation

The accuracy of the SPM configuration can be validated using the web app. Use the following procedure to validate the SPM configuration:

1. Open the web app.
2. Select a menu on the left of the page.

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 web app will verify the configuration of SPM and return a dialog box with one of two results:

- web app verified the SPM configuration successfully and the dialog box indicates no errors were found.
- 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

This chapter describes SPM operations, and includes the following information:

Topics

- [Managing the SPM Service using the Command Line](#)
- [Accessing SPM Information using the Web App](#)
- [Storage Policy Manager Action Result Codes](#)
- [Storage Policy Manager Logging](#)

Managing the SPM Service using the Command Line

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\filename.log*

Accessing SPM Information using the Web App

SPM information for objects can be viewed, and failed actions can be rescheduled using the Content Conductor web app. This section describes how to use the web app to access the information and what information is available through this interface.

Start the Content Conductor web app, 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 page by clicking on a listed bucket. This information is available for Disk Arrays on the Configuration > Resources > Disk Storage > Arrays page by clicking on a listed array.

If a newly archived object does not 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 page.

The ID column on the screen 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 Content Conductor 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 Content Conductor Request ID.

The remaining columns on this screen 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 will not appear.

Note: If the action is aging, the Content Conductor 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.

Filtering the SPM Actions Page View

The SPM Actions panel enables filtering which actions to view. Click the filter icon at the top of the screen 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.

Use the following procedure to assign a Storage Plan to the selected objects:

1. Open the Content Conductor web app.
2. Navigate to the Content Management > Catalog Browsing page.
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 from 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 does not complete correctly, SPM will automatically reschedule the action for re-execution at some time in the future. Failed actions can also be manually rescheduled to run immediately using the following procedure:

1. Open the Content Conductor web app.
2. Navigate 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.

Marking Failed Actions Complete

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 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 Content Conductor 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. See the Content Conductor API documentation on the Content Conductor Technical Support site for detailed information.
 - 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 Content Conductor 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. See the Content Conductor API documentation on the Content Conductor Technical Support site for detailed information.

- **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_CANNOT_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_CANNOT_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_CANNOT_SEND_MORE_REQUESTS
The Manager Monitor does not 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 Content Conductor, 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`.

Storage Policy Manager Logging

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.

Frequently Asked Questions

- What should be done if no SPM actions are being generated?
 - Confirm the SPM Service is operational.
 - Check the SPM configuration file in the conf/spm folder and verify that the values for POSTGRES_USER, POSTGRES_PASSWORD, and POSTGRES_CONNECTION parameters are set correctly.
 - Confirm that slots are enabled.
- What should be done if only Storage actions are being generated?
 - Confirm that the non-Storage type slots are enabled and are included in the correct Storage Plan.
 - Confirm that in the SPM Slot configuration on the Configuration > Resources > Storage Plans > Slots page that the Object Name, Object Source, and Object Collection filters are set so that they do not filter out the SPM actions you are expecting.
 - Check the SPM configuration file and confirm that DIVA_RESTORE_MAX_REQUESTS, DIVA_TRANSCODE_ARCHIVE_MAX_REQUESTS, or DIVA_METADATA_ARCHIVE_MAX_REQUESTS are not set to 0.
- What should be done if SPM actions are generated but are never executed (they stay in the Scheduled state)?

Check the Archive Date of the objects for which the actions were generated and compare it to the values configured in the Slot Begin Time and Slot End Time parameters, and Job Execution Begin Time and Job Execution End Time in the Web App to verify if one of the following cases is true:

- If the age of the object has not met or exceeded the number of minutes specified in the Slot Begin Time. In this case, the actions may still be executed but only after the age of the object has met or exceeded the number of minutes specified in the Slot Begin Time.
- If the age of the object has met or exceeded the number of minutes specified in Slot Begin Time and has not exceeded the minutes specified in the Slot End Time, but the clock time the object has existed (for example, 9:30 to the current time of 13:20) has not yet fallen within the Job Execution Begin Time and Job Execution

End Time range (for example, 13:30 to 15:30), the actions may still be executed, but only after the clock time has met or exceeded the Job Execution Begin Time.

- If the age of the object has exceeded the number of minutes specified in the Slot End Time without the SPM action being executed, then this SPM action will never execute and stay in the SCHEDULED state. For example, if the Slot End Time is 10 minutes, but the age of the object is 20 minutes, the SPM action for this object will not execute even if the clock time is within the slot Job Execution Begin Time and Job Execution End Time range because the object has expired.

This can occur by design where the object's lifetime is expected to expire before the slot Request Execution Begin Time is even reached. It can also occur if the Manager is frequently or constantly running more jobs than specified in the `DIVA_MANAGER_MONITOR_MAX_REQUESTS` parameter in `SPM.settings.json`. This, in turn, can cause the SPM Service to hold off submitting the SPM actions to the Manager to the point where the object eventually expires before its associated SPM action is allowed to be submitted to the Manager.

- What should be done if SPM actions always Fail Long?

Usually when an SPM action Fails Long it is due to the job submitted by SPM terminating in the Manager. Check the reason why the jobs are terminating in the Manager and remedy the problem there.

- Why are SPM actions removed from the SPM Actions List?

SPM actions are removed from the SPM Actions List when the objects they are associated with have been deleted from Content Conductor.

- Why does the Start At time not update when an SPM action is rescheduled?

The Start At time only represents the initial date and time when the action was initially executed.

- Why are copies being recreated on the disk after a Delete Instance?

Check that the Once Only flag in the slot configuration is set to true. If Once Only is set to false, then the system will continue to make copies of deleted instances.

- What are the different components of SPM?

- DSM (Disk Space Monitor) Controller

This module monitors the Watermarked array and does the cleanup when the High Watermark is reached.

- SPM Controller

This module is responsible for creating, updating, and executing actions based on the configuration.

- It appears that sometimes not all components start as part of SPM. What, if anything, controls whether a given component starts, and how do I tell if the DSM Controller is started since it is now integrated with SPM?

The SPM Controller is always started. The DSM Controller is started only if there is at least one Watermarked array configured.

- SPM appears to communicate only with the Postgres Database and the Manager (through Content Conductor API calls). Does SPM interact with any other processes?

SPM only interacts with the Postgres database and the Manager using the Content Conductor API. SPM may interact with Content Conductor Connect instead of Content Conductor Manager if configured to do so. Otherwise, SPM does not interact with any other processes.

- Is SPM event driven, or does it purely repeat some looped processing of slots and rules or database query results?

Only the DSM Controller part of SPM is event driven. It waits for the Watermarked array to reach the High Watermark to trigger the cleanup of the array.

- Is SPM stateful across restarts?

SPM is stateful across restarts. It saves the current state of the actions in the database. The actions that were running are recovered during the next SPM start.

- An archived object can only belong to a single Storage Plan. When an object is first archived, Content Conductor appears to archive to the default media for the corresponding Storage Plan, or the next-best media if the default media is not available for some reason.

Yes, an object can only belong to one Storage Plan. If a Storage Plan is specified in the job, that object is always assigned to that Storage Plan.

If the Storage Plan is specified and no target media is specified in the job, the object is assigned the given Storage Plan and archived to the default media of the Storage Plan. However, if the default media is not available, the object is then archived to the first available media of the Storage Slots in the given Storage Plan in ascending order of Slot Priority, Slot Window Begin Time, and Slot ID. If no media is available for the given Storage Plan the job is terminated.

- What are the meanings of the DATEARCHIVED, DATELASTUPDATE, and DATENEXTUPDATE properties for objects under SPM?

- DATEARCHIVED

This property is the date when object was archived.

- DATELASTUPDATE

This property is the date of the last time actions were created or updated for this object.

- DATENEXTUPDATE

This property is the date when SPM checks for any changes in the slot that will affect any actions of this object, and updates them if any changes are identified.

- A newly-archived object appears to have its DATELASTUPDATE and DATENEXTUPDATE fields set to the same value as DATEARCHIVED automatically by Content Conductor independent of SPM.

Yes, DATELASTUPDATE and DATENEXTUPDATE will change after SPM generates actions for that object.

Default Configuration Files and Parameters

This appendix describes SPM configuration parameters, and sample configuration files. The following information is included:

Topics

- [SPM Configuration Parameters](#)
- [SPM Default Configuration File \(SPM.settings.json.ini\)](#)

SPM Configuration Parameters

The following sections describe the default SPM parameters.

Storage Plan Definitions

- **Storage Plan Name**
This parameter is the name given to the Storage Plan.
- **Allow Last Instance Deletion**
 - Y deletes the object.
 - N only deletes the instance.
 - This parameter can be overridden in the configuration file.
 - The Disc Cleaning strategy is not applicable when set to Y.

Mediums Definitions

- **Medium Name**
This parameter is the given medium name (it is better if the same name as the Content Conductor Tape Group/Array Name is provided).
- **Storage Name**
This parameter is the name of the Storage Manager. This value is not used and Telestream recommends keeping the default value (DIVA).
- **Tape Group/Array Name**
This parameter is the Content Conductor Tape Group or Array name.
- **Watermarked**
This parameter is only applicable to Storage slots, and only for disk arrays configured as mediums. It monitors the medium's use (available space, used space, and so on). Select Y (applies watermarking), N (does not apply watermarking), or M (a combination of watermarking that is event dependent) to specify whether to apply watermarking to this medium. See [Actions, Action Steps, and Action States](#), and [Watermark Based Disk Cleaning Management](#) for detailed information on watermarking.

In general Telestream recommends that slots are not deleted, but rather disabled. Deleting a slot does not remove the content from the Content Conductor system.

For Storage slots, setting the Slot End Time to 0 forces SPM to delete any content created by this slot, and after all of the content is removed, the Storage slot can be deleted or disabled. Retaining the content in a Storage slot, when no further processing of future actions is required, disabling the slot instead of deleting it is the safer way to do this.

See [Watermark Based Disk Cleaning Management](#) for details on watermarking.

- Disc Cleaning Strategy
This parameter is the disc cleaning strategy to use:
 - Last Access Time
 - Size

Object Slot Definitions

- Slot Name
This parameter is the name given to the slot.
- Storage Plan
This parameter is the name given to the Storage Plan.
- Id
This parameter is automatically generated by SPM and not editable.
- Medium Associated With Slot
This parameter is the medium associated with the slot.
- Slot Request Type (or Slot Type)
This parameter is the type of job as follows:
 - Storage
A Copy job is initiated at the beginning of the slot, and then a Delete job is initiated at the end of the slot.
 - Transcode Archived
This job is initiated at the beginning of the slot.
 - Metadata Archive
This job is initiated at the beginning of the slot.
 - Restore
This job is initiated at the beginning of the slot.
- Slot Begin Time
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 original object is archived.
- Slot End Time
This parameter is the time (in minutes) from the object creation, and points to the slot end.
- Request Execution Begin Time
This parameter is the time in HH:MM format (00:00 to 24:00), and points to the time of the day when execution can start.

- Request Execution End Time
This parameter is the Time in HH:MM format (00:00 to 24:00), and points to the time of the day after which execution cannot start.
If the Request Execution Begin Time and Request Execution End Time are 00:00 and 24:00 respectively, there are no limitations on execution time. If the begin time is greater than end time, then the execution interval is taken from the begin time of the first day to the end time of the second day (including midnight).
- Request Priority
This parameter is the priority of operations for SPM and the Manager.

Filter Definitions

- Filter Name
This parameter is the name you want to call the filter.
- Id
This parameter is automatically generated by SPM and not editable.
- Medium Filter
This parameter is the name of the media to monitor.
- Object Collection Filter
This parameter is the Object Collection to display. Leaving this field blank performs no filtering and all objects are displayed regardless of the Collection. Any combination can be used except abc*def.
- Object Source Filter
This parameter is the source to display. Leaving this field blank performs no filtering and all objects are displayed regardless of the source. Any combination can be used except abc*def.
- Object Name Filter
This parameter is the Object Name to display. Leaving this field blank performs no filtering and all objects are displayed regardless of the name. Any combination can be used except abc*def.
- Minimum Object Size
This parameter is the minimum object size. A value of zero indicates no restrictions from the Maximum Object Size.
- Maximum Object Size
This parameter is the maximum object size. A value of zero indicates no restrictions from the Minimum Object Size.
- Size Units
This parameter is the object size units filter (KB, MB, GB).
- Storage Plan
This parameter is the name of the Storage Plan to filter.

SPM Actions Definitions

The following sections describe action definitions that are only used with Transcode Archived, Metadata Archive, and Restore slots.

Object Filters

- **Filter Name**
This parameter is the name to call the filter.
- **Storage Plan**
This parameter is the name of the Storage Plan to filter.
- **ID**
This parameter is automatically generated by SPM and is not editable.
- **Request Type**
This parameter is only used with Transcode Archived, Metadata Archive, and Restore slots.
- **Object Collection Filter**
This parameter is the Object Collection to display. Leaving this field blank indicates this parameter is not usable. Any combination can be used except abc*def.
- **Object Source Filter**
This parameter is the source to display. Leaving this field blank indicates this parameter is not usable. Any combination can be used except abc*def.
- **Object Name Filter**
This parameter is the Object Name to display. Leaving this field blank indicates this parameter is not usable. Any combination can be used except abc*def.

Transformation Rules

The following rules are only used with Transcode Archived and Metadata Archive slots.

- **Transcode Archived Request**
 - **Target Name**
This parameter is the name of the transcoded object. The ^ character results in using the original Object Name, and can be added to. For example, ^.txt.
 - **Target Collection**
This parameter is the collection of the transcoded object. The ^ character results in using the original Object Collection, and can be added to. For example, ^.txt.

- Metadata Archive Request
 - Target Name

This parameter is the name of the transcoded object. The ^ character results in using the original Object Name, and can be added to. For example, ^.txt.
 - Target Collection

This parameter is the collection of the transcoded object. The ^ character results in using the original Object Collection, and can be added to. For example, ^.txt.
- Restore Request
 - Target Name

Always leave this parameter set to the default ^ character.
 - Target Collection

Always leave this parameter set to the default ^ character.

Request Templates

- Request Type

This parameter is only used with Transcode Archived, Metadata Archive, and Restore jobs.

Transformation Rules

This parameter indicates to use an existing transformation rule.

- Id

This parameter is automatically generated by SPM and not editable.
- Object Unmanaged Storage Repository

For Metadata Archive this parameter is the Server to archive. For Restore this parameter is the Server to restore.
- Pathroot

This parameter is the Pathroot to use for Metadata Archive or Restore.
- Request Options

This parameter is any additional options necessary for Transcode Archived, Metadata Archive, or Restore.
- Request Comments

This parameter is comments for Transcode Archived, Metadata Archive, or Restore.
- QOS

This parameter is the Content Conductor Quality of Service.
- Cascade Delete Links

This parameter is only used for Transcode Archived and Metadata Archive.
- Cascade Restore Links

This parameter is only used for Transcode Archived and Metadata Archive.

SPM Default Configuration File (SPM.settings.json.ini)

Note: Parameters previously included in the spm.trace.ini have been moved to the SPM.settings.json.ini file. The spm.trace.ini file is no longer used.

The SPM.settings.json.ini file is a default configuration file and must be renamed to SPM.settings.json before use.

```
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information",
      "System.Net.Http.HttpClient": "None",
      "DIVACore.Common.Services.GatewayDiscoveryHostedService":
"None"
    },
    "Configuration": {
      "RetentionDays": 7
    }
  },
  "Version": "0.0.0.0",
  "AllowedHosts": "*",
  "DatabaseSettings": {
    "Databases": [
      {
        "ProfileName": "PostgresDatabase",
        "DatabaseName": "diva",
        "DatabaseType": "Postgres",
        "ConnectionString": "Host=127.0.0.1;Port=5432;",
        "User": "diva",
        "Password": "lib5"
      }
    ]
  },
  "ServiceSettings": {
    "DIVA_MANAGER_MONITOR_MAX_REQUESTS": 250,
    "DIVA_DELETE_MAX_REQUESTS": 300,
    "DIVA_RESTORE_MAX_REQUESTS": -1,
    "DIVA_TRANSCODE_ARCHIVE_MAX_REQUESTS": -1,
    "DIVA_METADATA_ARCHIVE_MAX_REQUESTS": -1,
    "DIVA_MANAGER_MONITOR_DELAY": 30,
    "DIVA_MANAGER_MONITOR_ACTION_DELAY": 60,
    "ACTION_SHORT_FAILURE_LIMIT": 5,
    "ACTION_STEP_SHORT_FAILURE_LIMIT": 2,
    "ACTION_STEP_WATCH_DELAY": 10,
    "ACTION_QUEUE_SIZE": 100,
    "SPM_EXECUTION_THREAD_REST": 5,
    "SPM_UPDATE_THREAD_REST": 20,
    "SPM_LOAD_THREAD_REST": 20,
    "RECOVER_ACTIONS_CHUNK": 500,
    "UPDATE_ACTIONS_CHUNK": 50,
  }
}
```

```
"GET_ACTIONS_CHUNK": 25,
"UPDATE_ACTIONS_NEXT_UPDATE": 20160,
"UPDATE_ACTIONS_RETRY_FAILED_DELAY": 720,
"ALLOW_OBJECT_DELETION": 0,
"DELETE_OBJECT_ONLY_LAST_INSTANCE": 0,
"METADATA_ARCHIVE_TRANSFORMED_OBJECT_NAME": 1,
"DSM_DIW_REQUEST_PRIORITY": 15,
"DSM_SPACE_MONITOR_THREAD_REST": 10,
"DSM_ACTIVATE_DIW_BY_SIZE_CHUNK": 50,
"DSM_ACTIVATE_DIW_BY_TIME_CHUNK": 50,
"DSM_ACTIVATE_ACTIONS_CHUNK": 50,
"TRACE_LEVEL": 1
},
"HttpServer": {
  "Endpoints": {
    "Http": {
      "Host": "localhost",
      "Port": 1976,
      "Scheme": "http"
    },
    "Https": {
      "Host": "localhost",
      "Port": 1977,
      "Scheme": "https",
      "FilePath": "../../security/certificates/SPM.p12"
    }
  }
},
"DiscoverySettings": {
  "Url": "https://127.0.0.1:8761/eureka/apps/",
  "AppName": "spm-service"
},
"DIVACoreAPISettings": {
  "Url": "https://127.0.0.1:8765/",
  "TimeoutInMs": 20000
}
}
```